Manifold Sampling for Piecewise Linear Nonconvex Optimization

Jeffrey Larson, Kamil Khan, Stefan Wild

Argonne National Laboratory

May 25, 2017

Problem statement

We are interested in solving the problem:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x) \triangleq \psi(x) + h(F(x))$$

where $\psi : \mathbb{R}^n \to \mathbb{R}$, $F : \mathbb{R}^n \to \mathbb{R}^p$, $h : \mathbb{R}^p \to \mathbb{R}$,

Problem statement

We are interested in solving the problem:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x) \triangleq \psi(x) + h(F(x))$$

where $\psi : \mathbb{R}^n \to \mathbb{R}$, $F : \mathbb{R}^n \to \mathbb{R}^p$, $h : \mathbb{R}^p \to \mathbb{R}$, and

- ψ is smooth with known derivatives

Problem statement

We are interested in solving the problem:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x) \triangleq \psi(x) + h(F(x))$$

where $\psi : \mathbb{R}^n \to \mathbb{R}$, $F : \mathbb{R}^n \to \mathbb{R}^p$, $h : \mathbb{R}^p \to \mathbb{R}$, and

- ψ is smooth with known derivatives
- h is nonsmooth, piecewise linear, and has a known structure (cheap to evaluate)

Problem statement

We are interested in solving the problem:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x) \triangleq \psi(x) + h(F(x))$$

where $\psi : \mathbb{R}^n \to \mathbb{R}$, $F : \mathbb{R}^n \to \mathbb{R}^p$, $h : \mathbb{R}^p \to \mathbb{R}$, and

- ψ is smooth with known derivatives
- h is nonsmooth, piecewise linear, and has a known structure (cheap to evaluate)
- F is smooth, nonlinear, and has a relatively unknown structure (expensive to evaluate)

Problem statement

We are interested in solving the problem:

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \quad f(x) \triangleq \psi(x) + h(F(x))$$

where $\psi : \mathbb{R}^n \to \mathbb{R}$, $F : \mathbb{R}^n \to \mathbb{R}^p$, $h : \mathbb{R}^p \to \mathbb{R}$, and

- ψ is smooth with known derivatives
- h is nonsmooth, piecewise linear, and has a known structure (cheap to evaluate)
- F is smooth, nonlinear, and has a relatively unknown structure (expensive to evaluate)

Piecewise linear h does not imply $h \circ F$ is piecewise linear.

Notes

- The *manifold sampling* framework does not require the availability of the Jacobian ∇F.

Notes

- The *manifold sampling* framework does not require the availability of the Jacobian ∇F.

- Applicable both when inexact values for $\nabla F(x)$ are available and in the derivative-free case, when only $F(x)$ is available.

Notes

- The *manifold sampling* framework does not require the availability of the Jacobian ∇F.

- Applicable both when inexact values for $\nabla F(x)$ are available and in the derivative-free case, when only $F(x)$ is available.

- We will build component models m^{F_i} of each F_i around points x. We can then use $\nabla M(x) \in \mathbb{R}^{n\times p}$ where

$$\nabla M(x) \triangleq \left[\nabla m^{F_1}(x), \dots, \nabla m^{F_p}(x)\right].$$

Piecewise linear functions

Definition

A function $h\colon \mathbb{R}^p \to \mathbb{R}$ is *piecewise linear* if h is continuous and there exists a finite collection $\mathfrak{H} \triangleq \{h_i : i = 1, \ldots, \hat{m}\}$ of affine functions that map $\mathbb{R}^p$ into $\mathbb{R}$, for which

$$h(z) \in \left\{\tilde{h}(z) : \tilde{h} \in \mathfrak{H}\right\}, \qquad \forall z \in \mathbb{R}^p.$$

- h is a *continuous selection* of $\mathfrak{H}$.
- Elements of $\mathfrak{H}$ are *selection functions of* h.
- $h_i : z \in \mathbb{R}^p \mapsto \langle a_i, z\rangle + b_i$ for each i.

Piecewise linear functions

Definition

A function $h\colon \mathbb{R}^p \to \mathbb{R}$ is *piecewise linear* if h is continuous and there exists a finite collection $\mathfrak{H} \triangleq \{h_i : i = 1, \ldots, \hat{m}\}$ of affine functions that map $\mathbb{R}^p$ into $\mathbb{R}$, for which

$$h(z) \in \left\{\tilde{h}(z) : \tilde{h} \in \mathfrak{H}\right\}, \qquad \forall z \in \mathbb{R}^p.$$

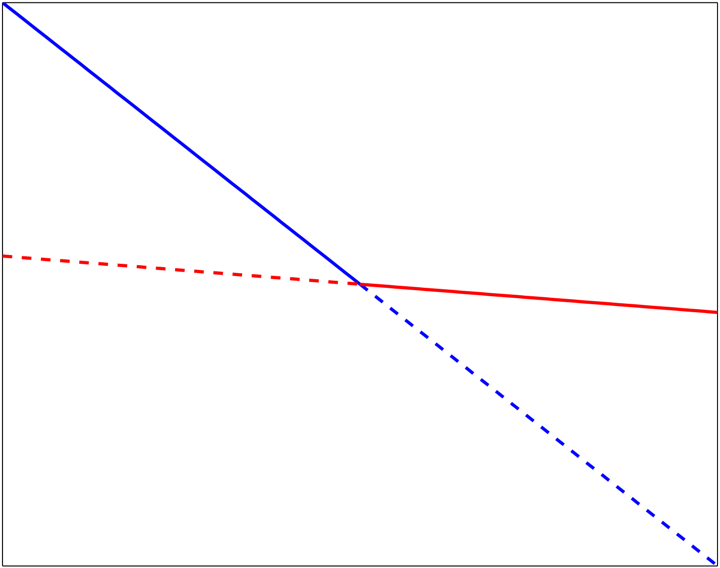
- h is a *continuous selection* of $\mathfrak{H}$.
- Elements of $\mathfrak{H}$ are *selection functions of* h.
- $h_i : z \in \mathbb{R}^p \mapsto \langle a_i, z\rangle + b_i$ for each i.

Definition

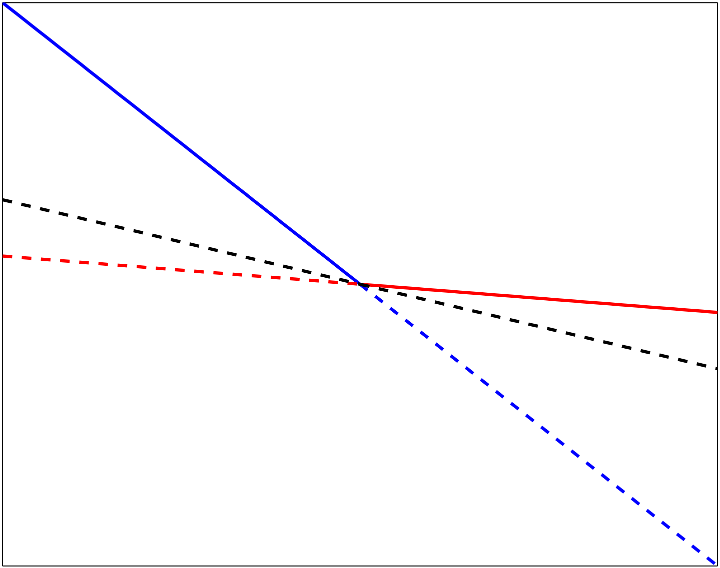
$$\mathcal{S}_i \triangleq \{y : h(y) = h_i(y)\}, \quad \tilde{\mathcal{S}}_i \triangleq \mathbf{cl}\left(\mathbf{int}\left(\mathcal{S}_i\right)\right), \quad I_h(z) \triangleq \left\{i : z \in \tilde{\mathcal{S}}_i\right\},$$

h_i for $i \in I_h(z)$ is an *essentially active selection function* for h at z.

Essentially active

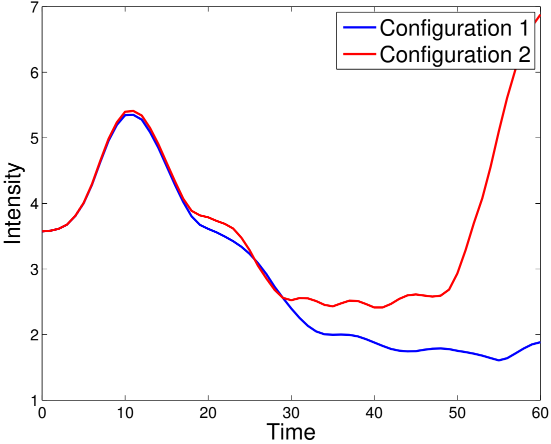


Essentially active



Laser pulse propagating in a plasma channel

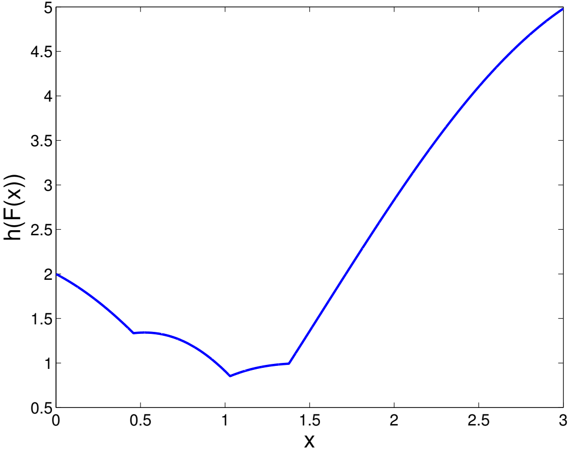
Determine plasma channel properties that minimize the maximum difference in the laser intensity.



$$f(x) = \max_{\Omega_1} \{F_i(x)\} - \min_{\Omega_2} \{F_i(x)\}$$

Formulation

$$h(F(x)) = \max\{\sin(2x)+1, \cos(2x), x\} - \min\{\sin(2x)+1, \cos(2x), x\}$$



A generalized derivative

Definition

The *generalized Clarke subdifferential* of f at x is defined as

$$\partial_{\mathrm{C}} f(x) \triangleq \mathbf{co}\left(\left\{\xi : \xi = \lim_{y^j \to x} \nabla f(y^j) : \; y^j \in \mathcal{D}\right\}\right),$$

where $\mathbf{co}\,(\cdot)$ denotes the convex hull.

A generalized derivative

Definition

The *generalized Clarke subdifferential* of f at x is defined as

$$\partial_{\mathrm{C}} f(x) \triangleq \mathbf{co}\left(\left\{\xi : \xi = \lim_{y^j \to x} \nabla f(y^j) :\; y^j \in \mathcal{D}\right\}\right),$$

where $\mathbf{co}\,(\cdot)$ denotes the convex hull.

For our case:

$$\partial_{\mathrm{C}} h(z) = \mathbf{co}\left(\{a_i : i \in I_h(z)\}\right)$$

A generalized derivative

Definition

The *generalized Clarke subdifferential* of f at x is defined as

$$\partial_{\mathrm{C}} f(x) \triangleq \mathbf{co}\left(\left\{\xi : \xi = \lim_{y^j \to x} \nabla f(y^j) : \; y^j \in \mathcal{D}\right\}\right),$$

where $\mathbf{co}\,(\cdot)$ denotes the convex hull.

For our case:

$$\partial_{\mathrm{C}} h(z) = \mathbf{co}\,(\{a_i : i \in I_h(z)\})$$

Definition

A point x is called a *Clarke stationary* point of f if $0 \in \partial_{\mathrm{C}} f(x)$.

Algorithm components

- Generator set $\mathfrak{G}^k$

Algorithm components

- Generator set $\mathfrak{G}^k$
- Smooth master model m_k^f

Algorithm components

- Generator set $\mathfrak{G}^k$
- Smooth master model m_k^f
- Trust-region subproblem solution s^k

Algorithm components

- Generator set $\mathfrak{G}^k$
- Smooth master model m_k^f
- Trust-region subproblem solution s^k
- Measuring decent with ρ_k

Generator set

At some iterate x^k,

$$\mathfrak{G}^k \triangleq \bigcup_{i \in I_h(F(x^k))} \left\{ \nabla \psi(x^k) + \nabla M(x^k) a_i \right\}$$

where $I_h(F(x^k))$ is the set of essentially active indices.

Generator set

At some iterate x^k,

$$\mathfrak{G}^k \triangleq \bigcup_{i \in I_h(F(x^k))} \left\{ \nabla\psi(x^k) + \nabla M(x^k) a_i \right\}$$

where $I_h(F(x^k))$ is the set of essentially active indices.

Or, given a set of points $Y = \left\{x^k, y^2, \ldots, y^p\right\} \subset \mathcal{B}(x^k, \Delta_k)$,

$$\mathfrak{G}^k \triangleq \bigcup_{y \in Y} \bigcup_{i \in I_h(F(y))} \left\{ \nabla\psi(x^k) + \nabla M(x^k) a_i \right\}$$

Generator set

At some iterate x^k,

$$\mathfrak{G}^k \triangleq \bigcup_{i \in I_h(F(x^k))} \{\nabla\psi(x^k) + \nabla M(x^k) a_i\}$$

where $I_h(F(x^k))$ is the set of essentially active indices.

Or, given a set of points $Y = \{x^k, y^2, \ldots, y^p\} \subset \mathcal{B}(x^k, \Delta_k)$,

$$\mathfrak{G}^k \triangleq \bigcup_{y \in Y} \bigcup_{i \in I_h(F(y))} \{\nabla\psi(x^k) + \nabla M(x^k) a_i\}$$

Assumption

The set $\mathfrak{G}^k$ satisfies

$$\{\nabla\psi(x^k) + \nabla M(x^k)\, a_i : i \in I_h(F(x^k))\} \subseteq \mathfrak{G}^k$$
$$\mathfrak{G}^k \subseteq \{\nabla\psi(x^k) + \nabla M(x^k)\, a_i : y \in \mathcal{B}\left(x^k; \Delta_k\right), i \in I_h(F(y))\}.$$

Smooth master model

Our model gradients around iterate x^k satisfy

$$g^k \triangleq \mathbf{proj}\left(0, \mathbf{co}\left(\mathfrak{G}^k\right)\right) \in \mathbf{co}\left(\mathfrak{G}^k\right),$$

Let λ^* be the corresponding coefficients so that $g^k = G^k \lambda^*$.

Smooth master model

Our model gradients around iterate x^k satisfy

$$g^k \triangleq \mathbf{proj}\left(0, \mathbf{co}\left(\mathfrak{G}^k\right)\right) \in \mathbf{co}\left(\mathfrak{G}^k\right),$$

Let λ^* be the corresponding coefficients so that $g^k = G^k\lambda^*$.

Define

$$A^k \triangleq \begin{bmatrix} | & & | \\ a_{j_1} & \cdots & a_{j_t} \\ | & & | \end{bmatrix},$$

and set $w^k = A^k\lambda^*$. Define the smooth *master model* $m_k^f : \mathbb{R}^n \to \mathbb{R}$,

$$m_k^f(x) \triangleq \psi(x^k) + \sum_{i=1}^{p} w_i^k m^{F_i}(x) + \sum_{i=1}^{p} \lambda_i^* b_{j_i}.$$

Trust region subproblem

Approximately solve

$$\begin{aligned} &\underset{s}{\text{minimize}}\ m_k^f(x^k+s) \\ &\textbf{subject to: } s \in \mathcal{B}(0, \Delta_k) \end{aligned}$$

to obtain a solution s satisfying

$$\psi(x^k) - \psi(x^k+s) + \langle M(x^k) - M(x^k+s), w^k \rangle \geq \frac{\kappa_{\mathrm{d}}}{2} \|g^k\| \min\left\{\Delta_k, \frac{\|g^k\|}{\kappa_{\mathrm{mh}}}\right\}.$$

Measuring descent

- Descent is measured using some selection function $h^{(k)}$ and not h

Measuring descent

- Descent is measured using some selection function $h^{(k)}$ and not h

- Must ensure information about $h^{(k)}$ is in $\mathfrak{G}^k$ before taking a step

Measuring descent

- Descent is measured using some selection function $h^{(k)}$ and not h
- Must ensure information about $h^{(k)}$ is in $\mathfrak{G}^k$ before taking a step
- $h^{(k)}$ must satisfy

$$h^{(k)}(F(x^k)) \leq h(F(x^k)) \qquad \text{and} \qquad h^{(k)}(F(x^k+s^k)) \geq h(F(x^k+s^k)),$$

Measuring descent

- Descent is measured using some selection function $h^{(k)}$ and not h

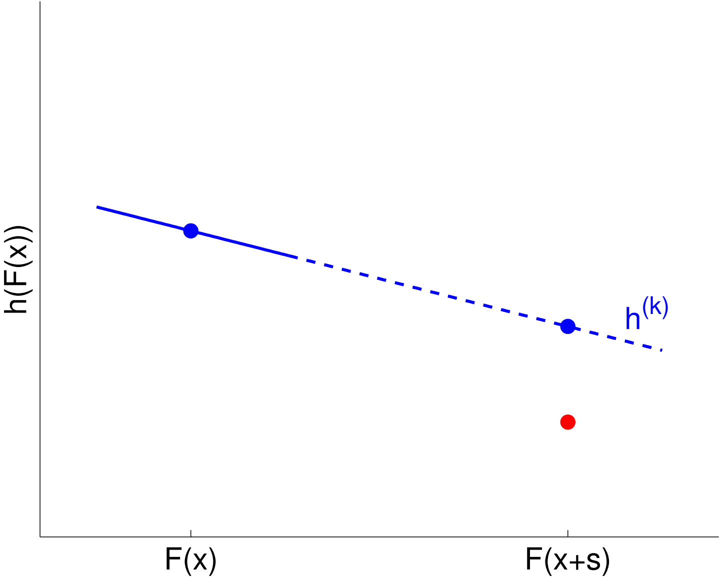
- Must ensure information about $h^{(k)}$ is in $\mathfrak{G}^k$ before taking a step

- $h^{(k)}$ must satisfy

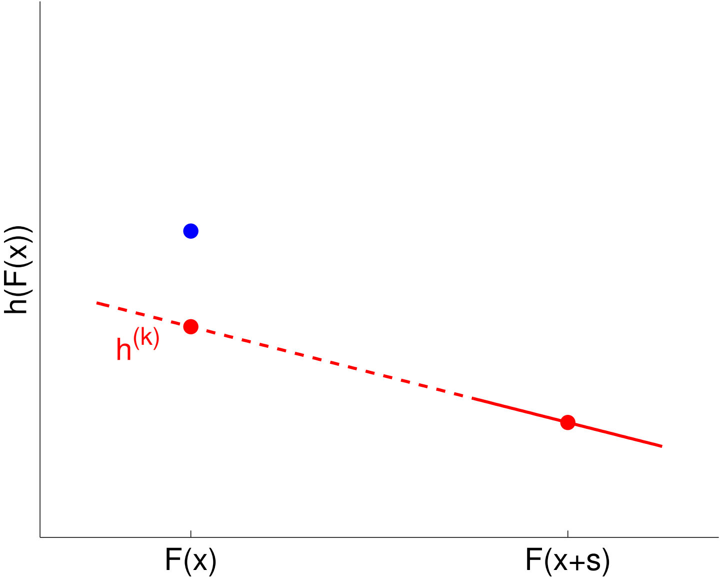
$$h^{(k)}(F(x^k)) \leq h(F(x^k)) \qquad \text{and} \qquad h^{(k)}(F(x^k+s^k)) \geq h(F(x^k+s^k)),$$

- $$\rho_k \triangleq \frac{\psi(x^k) - \psi(x^k+s^k) + h^{(k)}(F(x^k)) - h^{(k)}(F(x^k+s^k))}{\psi(x^k) - \psi(x^k+s^k) + \langle M(x^k) - M(x^k+s^k), a^{(k)} \rangle}$$

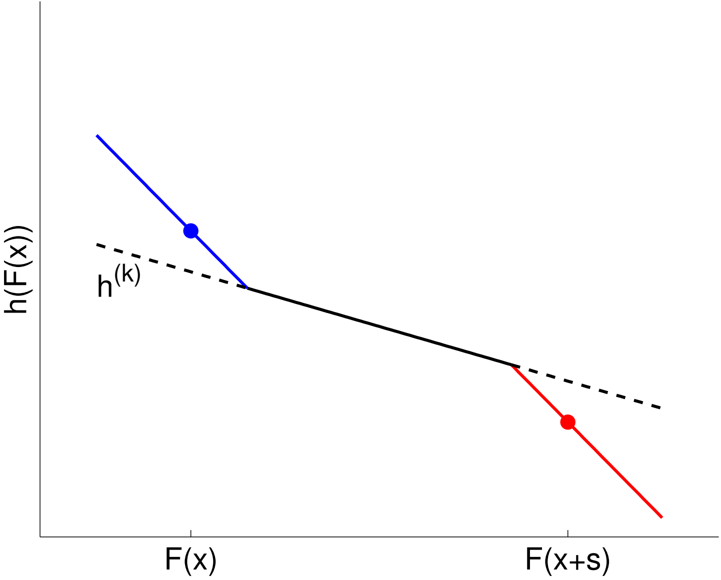
Examples of $h^{(k)}$



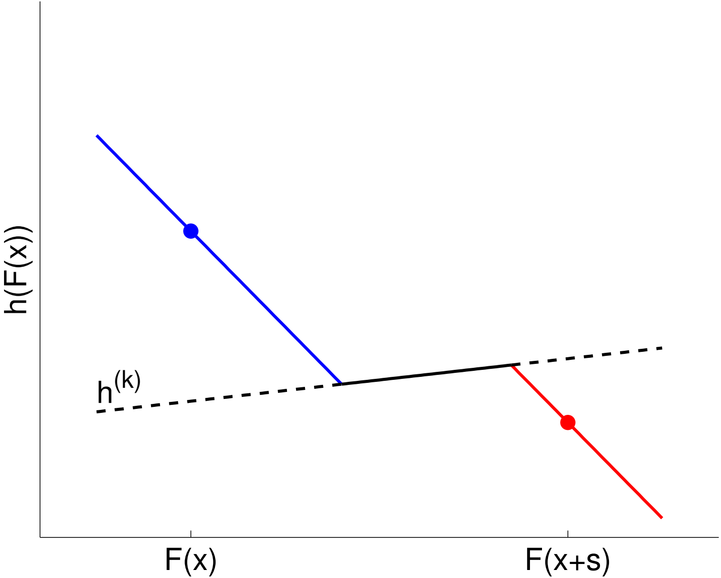
Examples of $h^{(k)}$



Examples of $h^{(k)}$



Examples of $h^{(k)}$



Algorithm components

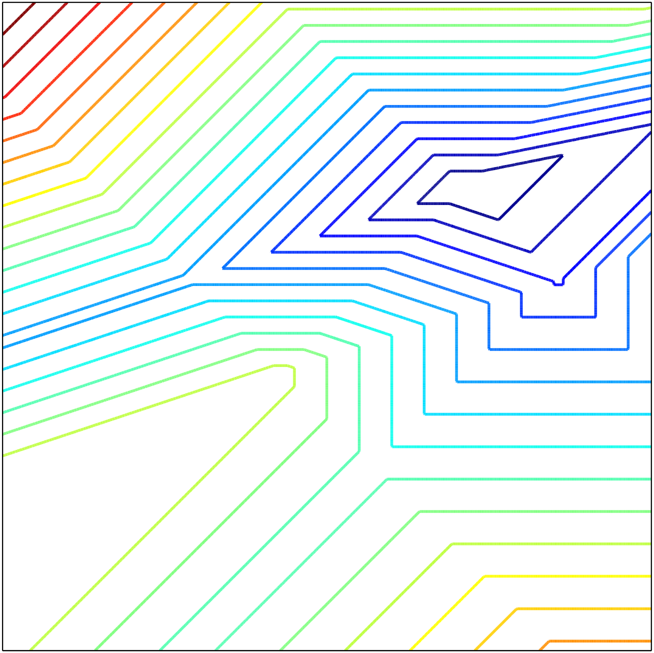
- Generator set $\mathfrak{G}^k$
- Smooth master model m_k^f
- Trust-region subproblem solution s^k
- Measuring decent with ρ_k

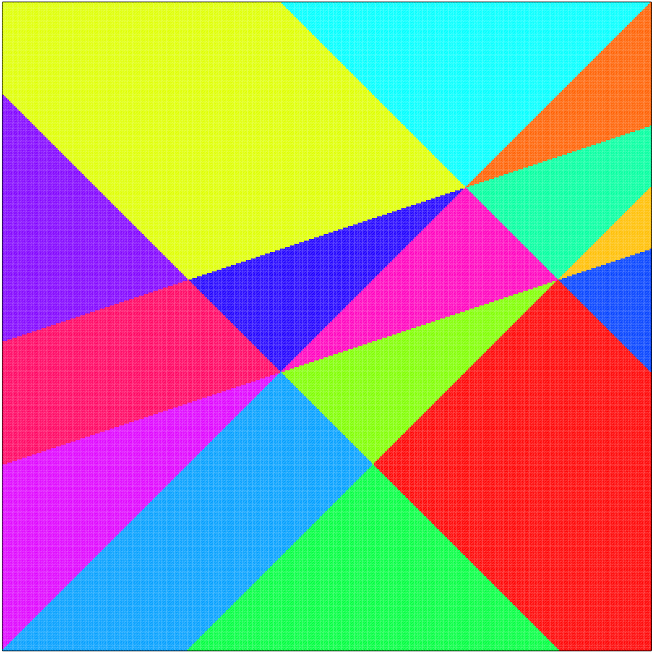
Algorithm MS4PL

Choose initial iterate x^0 and trust-region radius $\Delta_0 > 1$

for $k = 0, 1, 2, \ldots$ **do**

- Build p component models m^{F_i} that are fully linear on $\mathcal{B}(x^k, \Delta_k)$
- Form $\nabla M(x^k)$ using $\nabla m^{F_i}(x^k)$ and construct $\mathfrak{G}^k \subset \mathbb{R}^n$
- $\rho_k \leftarrow -\infty$
- **while** $\rho_k = -\infty$ **do**
  - Update component models m^{F_i}; build master model m^f
  - **if** $\Delta_k < \eta_2 \|\nabla m^f(x^k)\|$ (*acceptability* criterion) **then**
    - Approximately solve TRSP to obtain s^k
    - Evaluate $F(x^k + s^k)$ and find $h^{(k)}$
    - **if** $(\nabla \psi(x^k) + \nabla M(x^k)\, a^{(k)}) \in \mathfrak{G}^k$ **then**
      - Calculate ρ_k
    - **else**
      - $\mathfrak{G}^k \leftarrow \mathfrak{G}^k \cup \{\nabla \psi(x^k) + \nabla M(x^k)\, a^{(k)}\}$
  - **else**
    - break out of while-loop; iteration is *unacceptable*
- **if** $\rho_k > \eta_1 > 0$ (*successful* iteration) **then**
  - $x^{k+1} \leftarrow x^k + s^k$, $\Delta_{k+1} \leftarrow \min\{\gamma_{\mathrm{inc}}\Delta_k, \Delta_{\max}\}$
- **else**
  - $x^{k+1} \leftarrow x^k$, $\Delta_{k+1} \leftarrow \gamma_{\mathrm{dec}}\Delta_k$



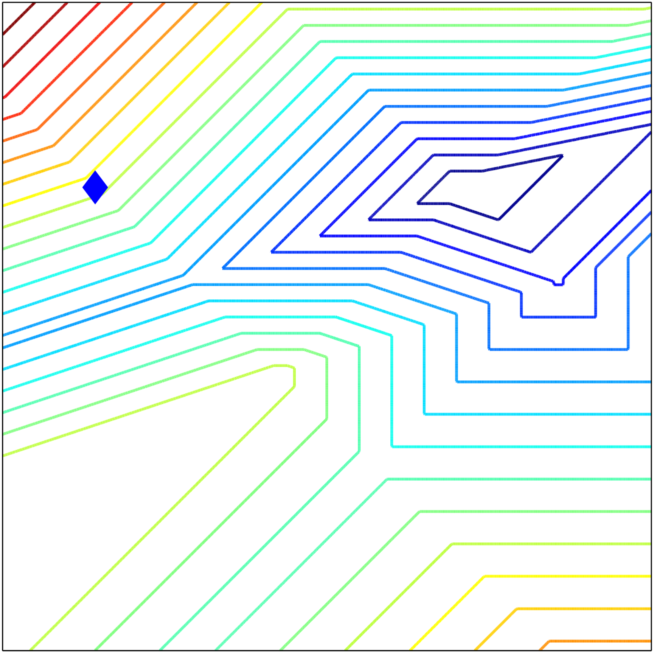


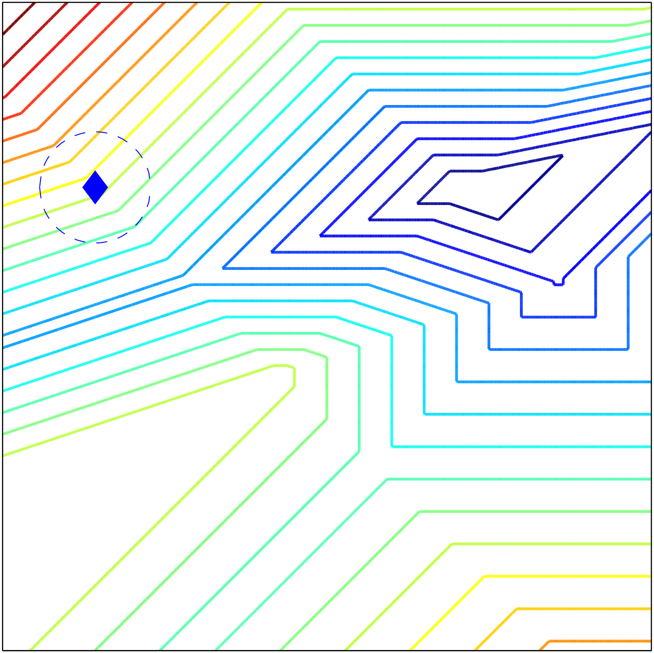
Generator set

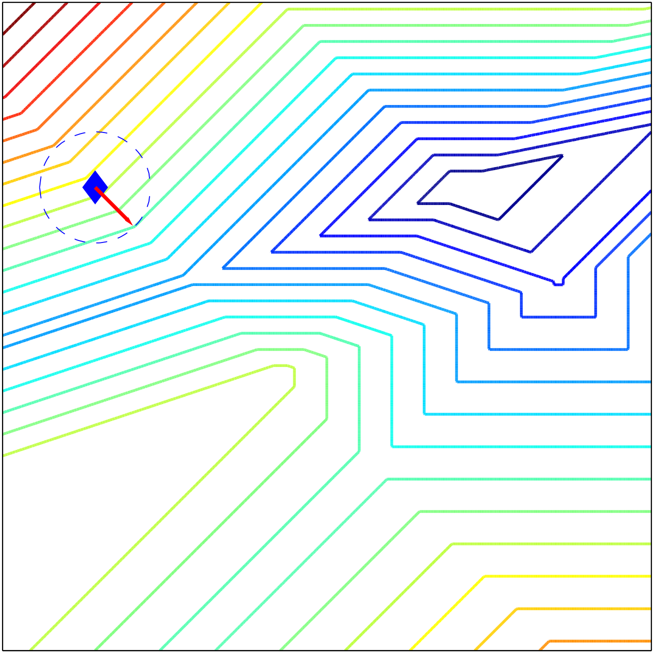
At some iterate x^k,

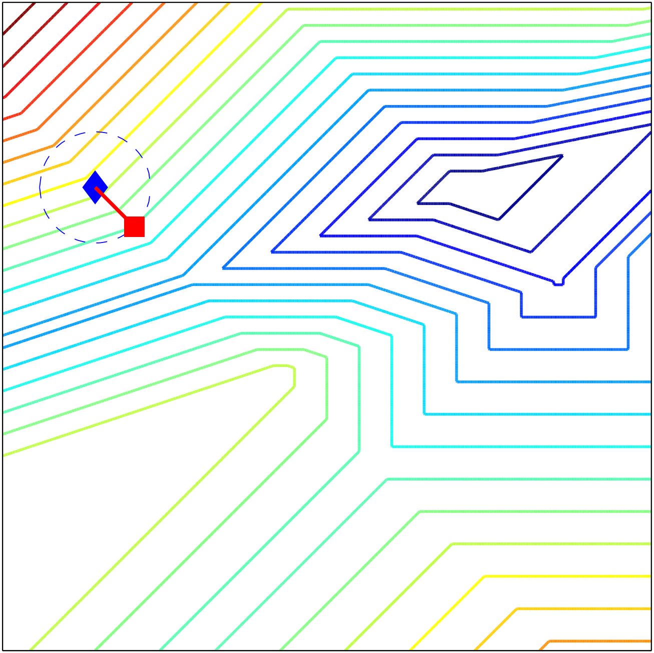
$$\mathfrak{G}^k \triangleq \bigcup_{i \in I_h(F(x^k))} \left\{ \nabla\psi(x^k) + \nabla M(x^k) a_i \right\}$$

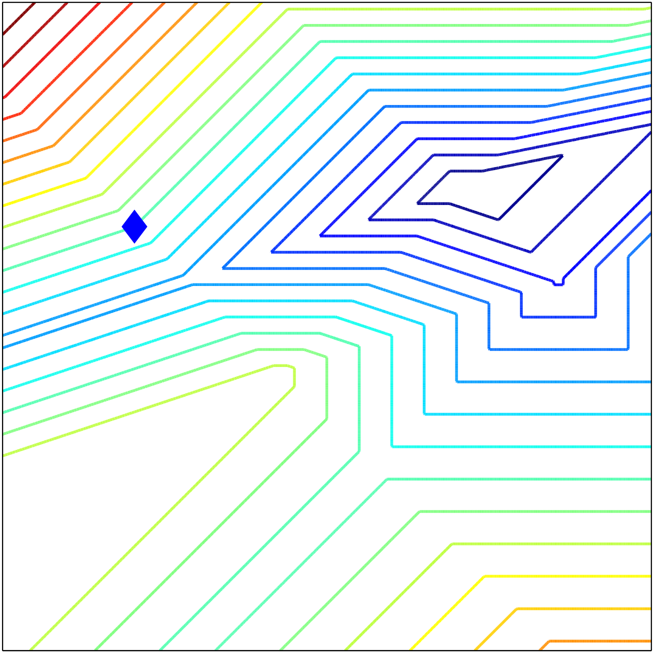
where $I_h(F(x^k))$ is the set of essentially active indices.

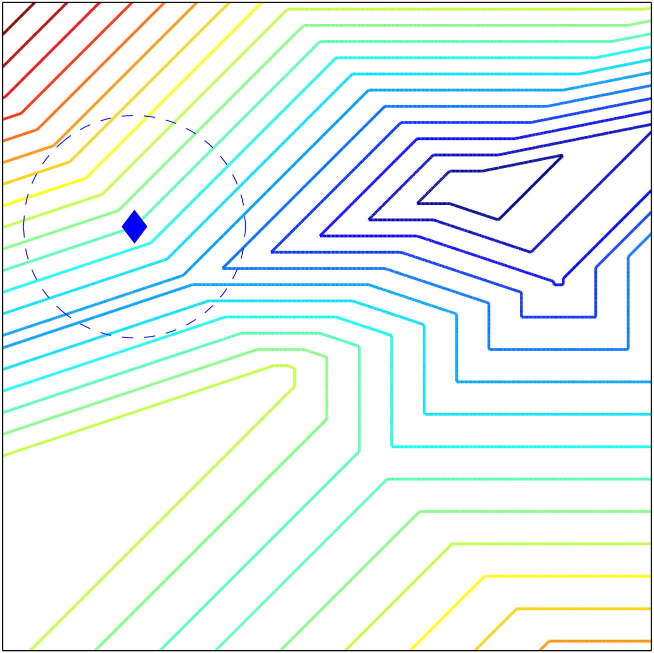


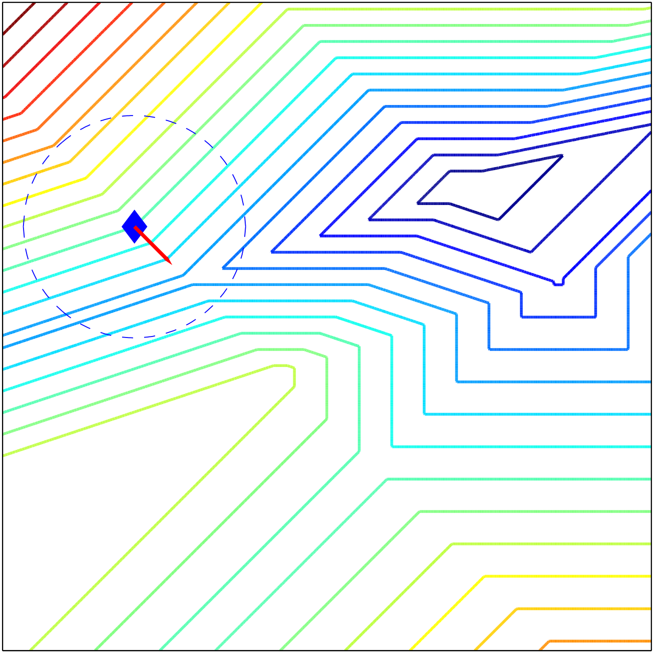


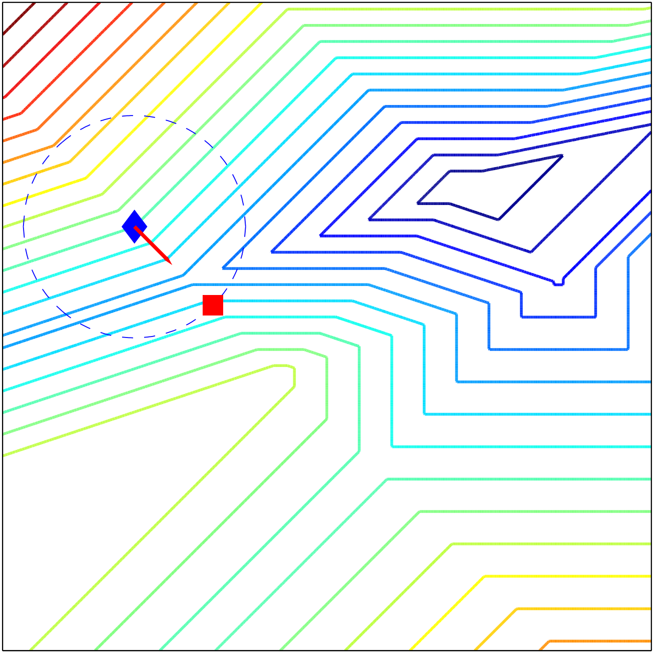


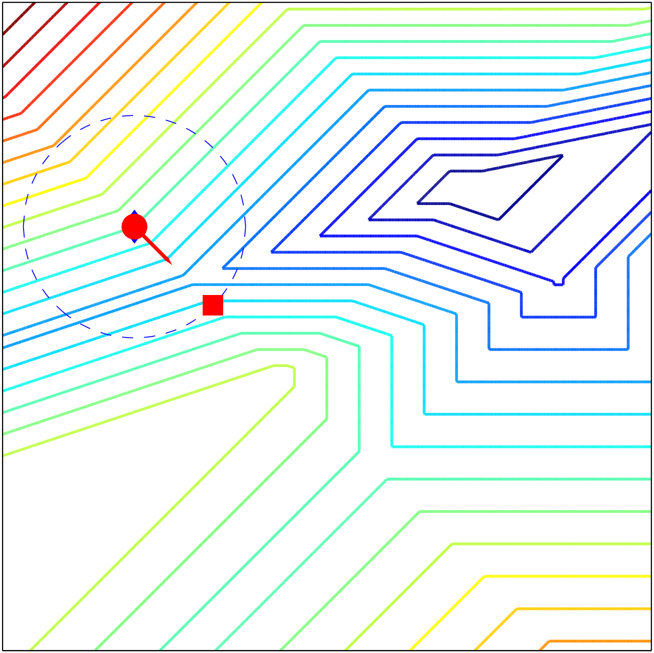


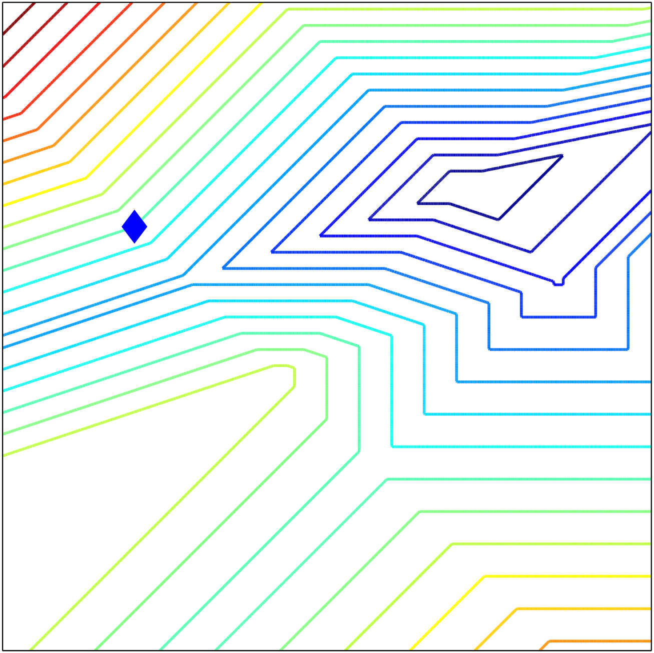


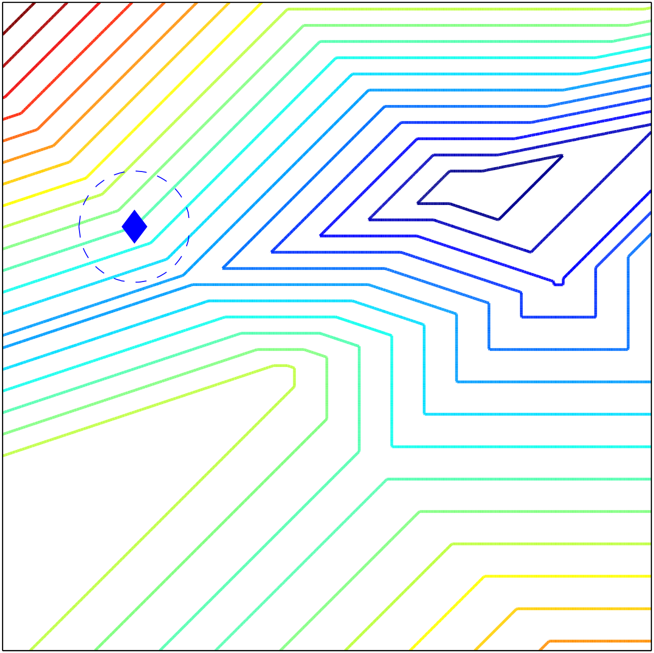


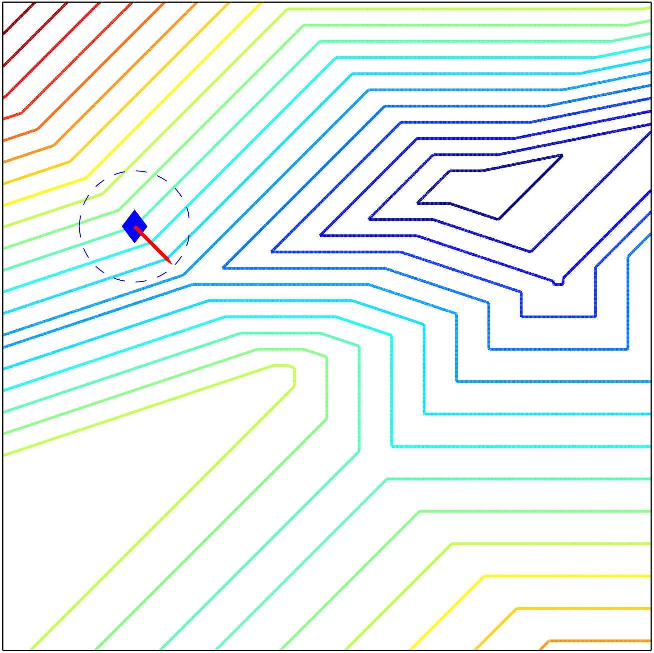


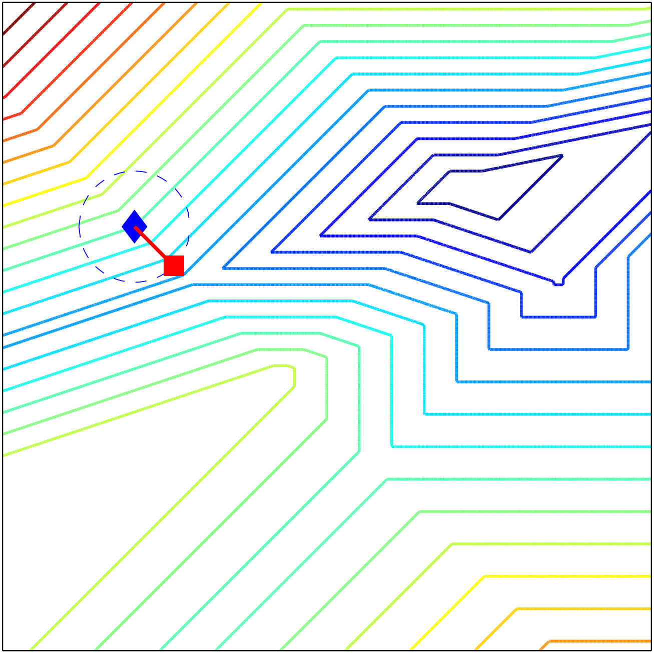


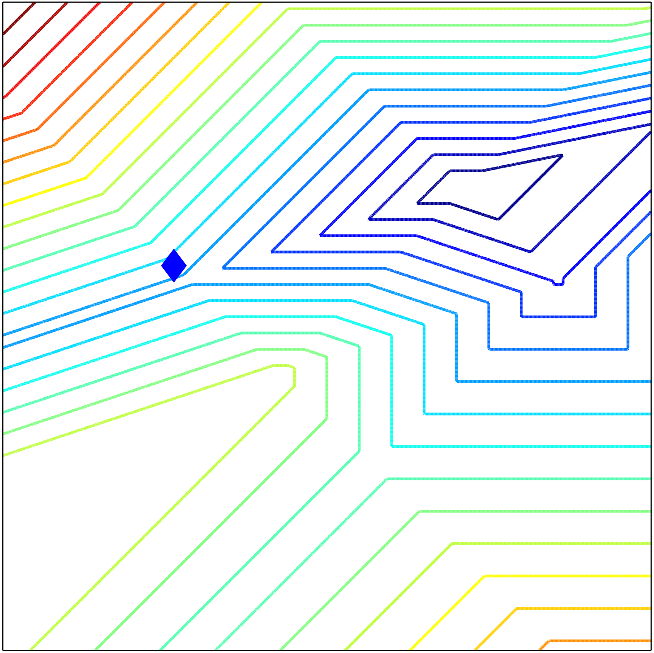


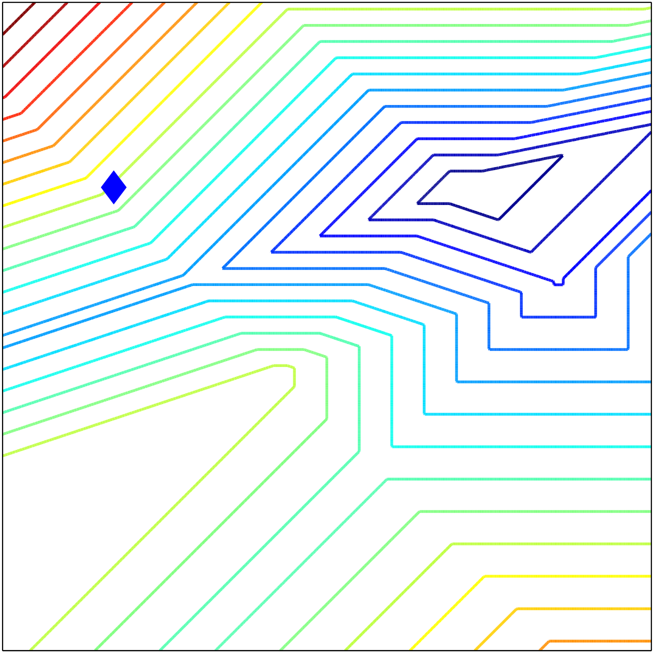


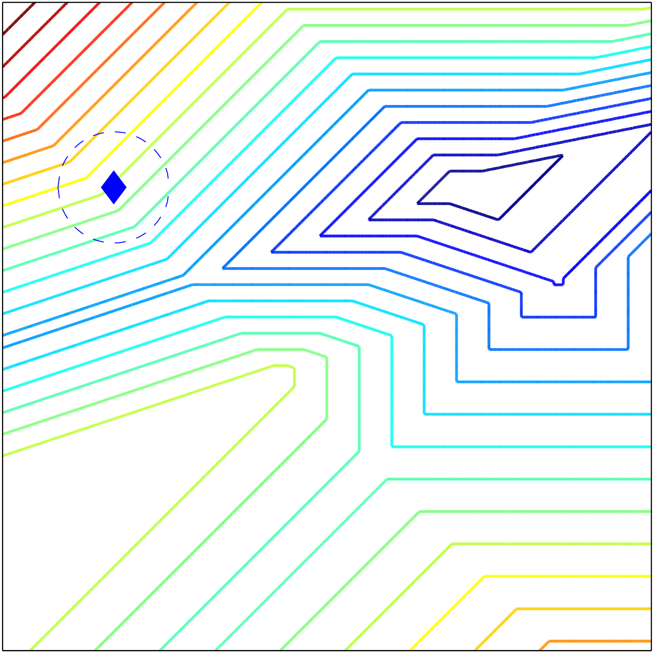


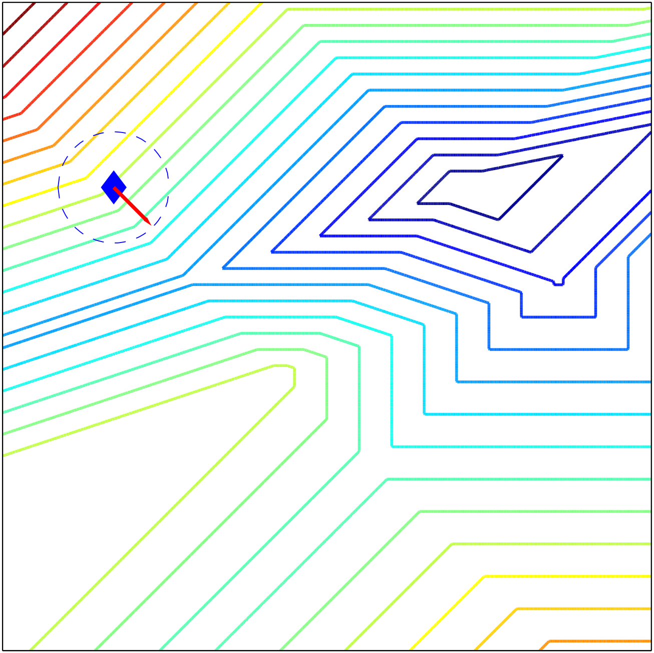


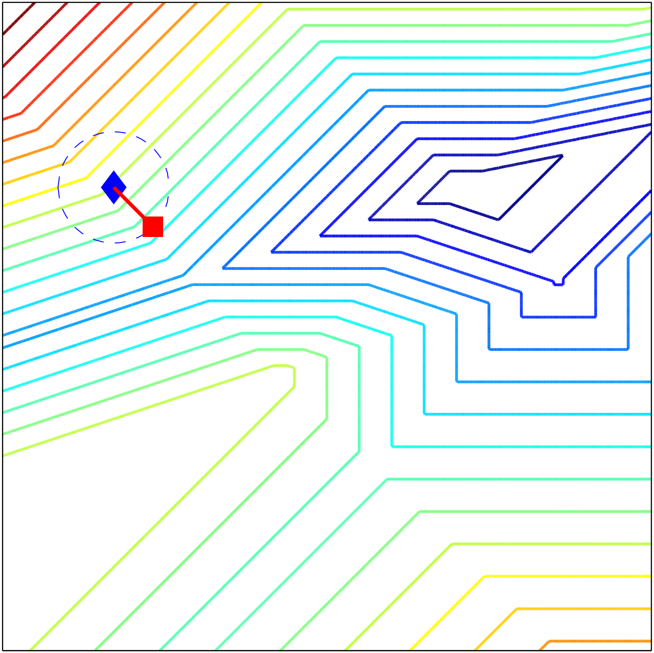


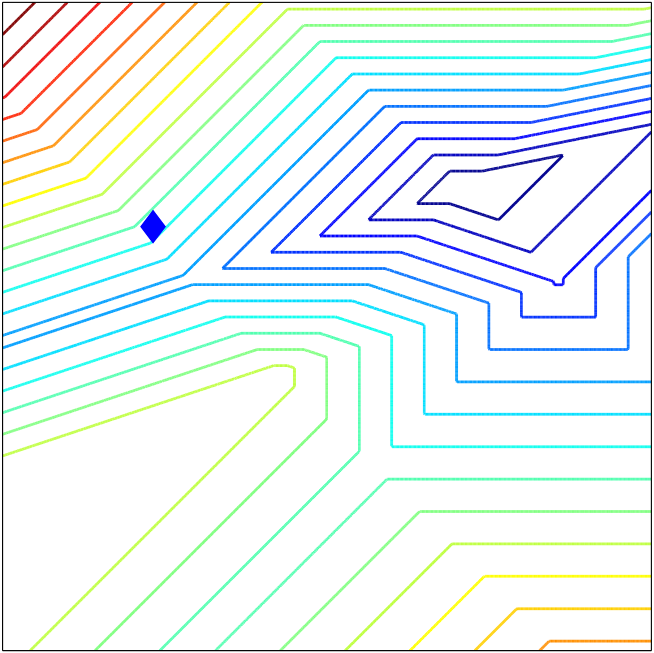


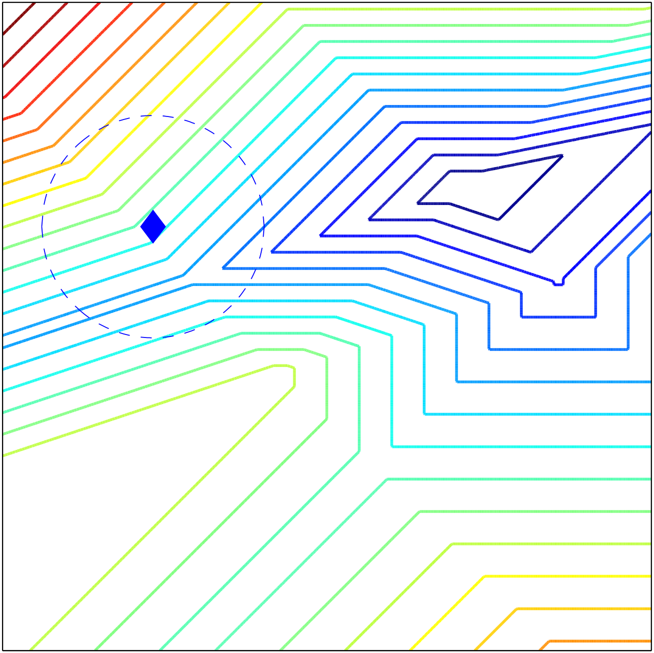


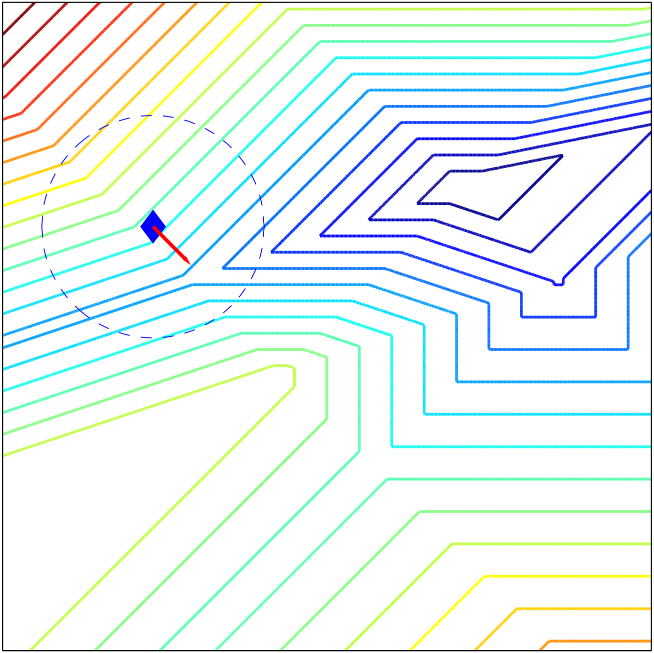


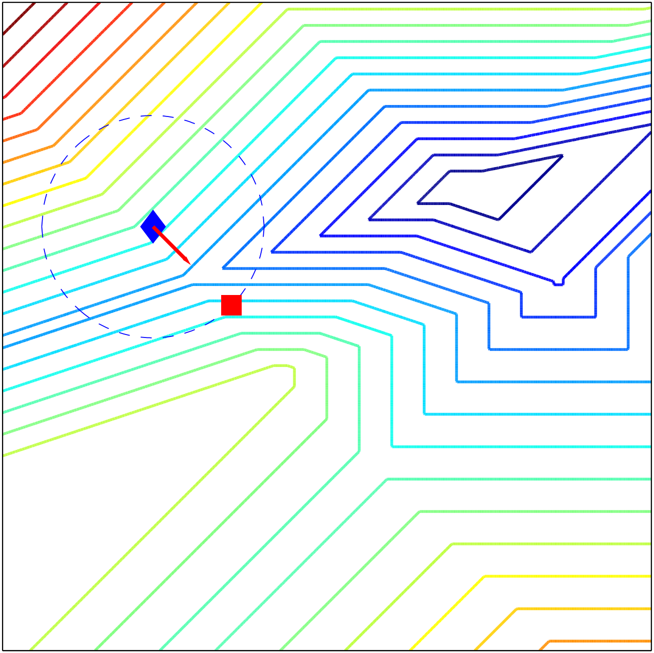


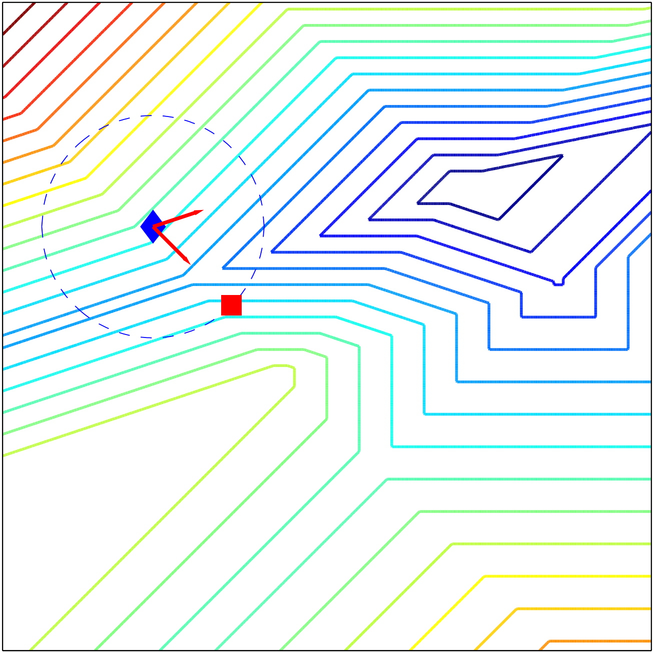


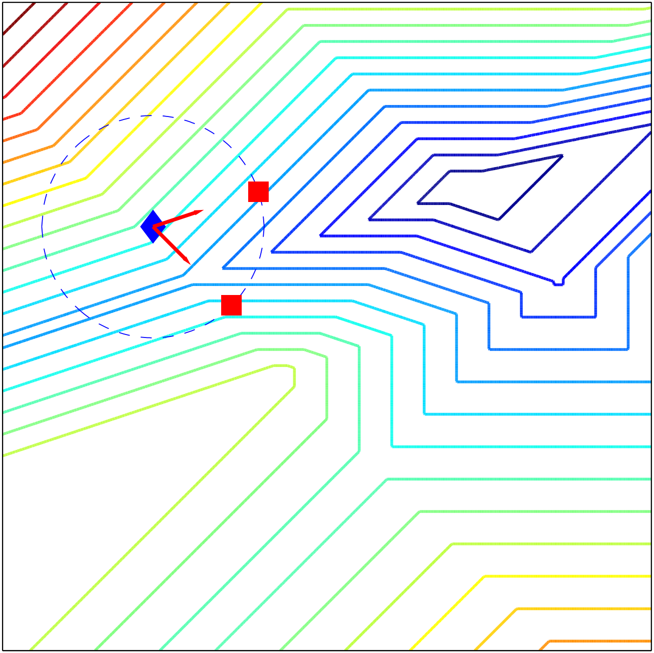


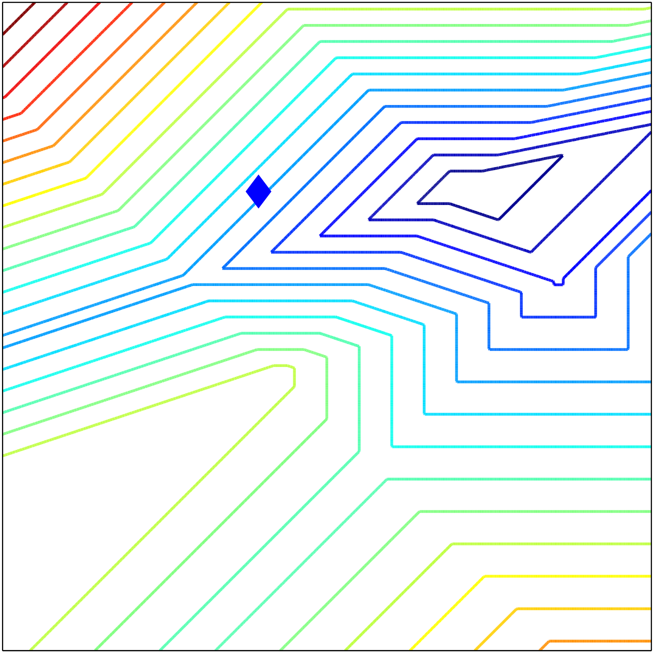


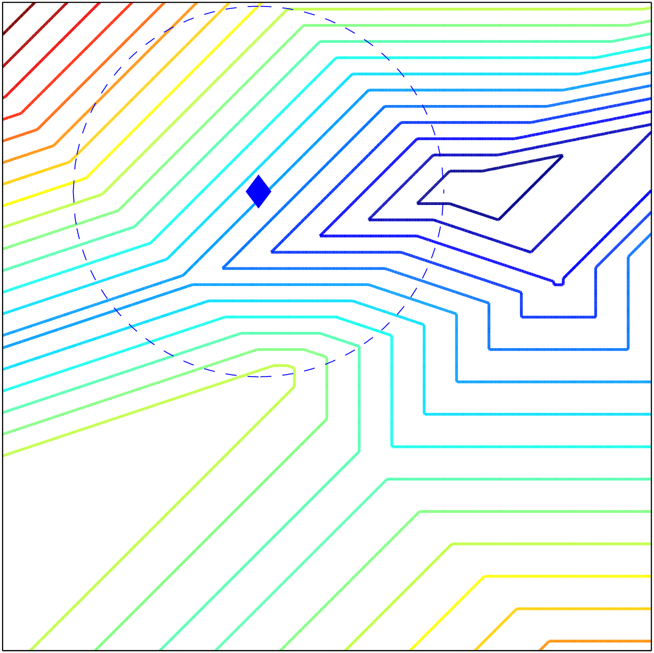


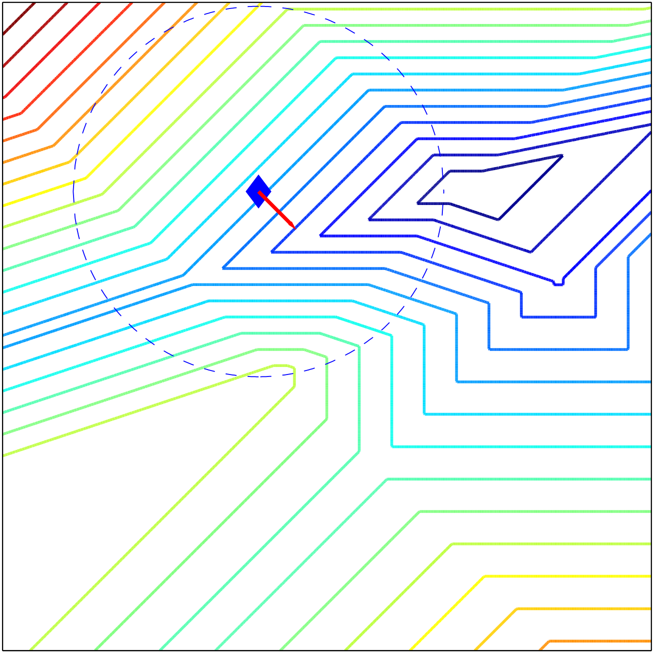


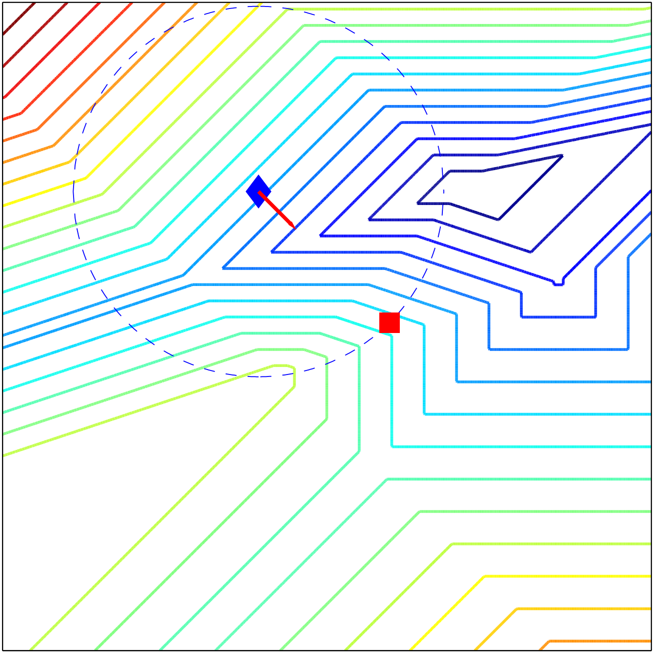


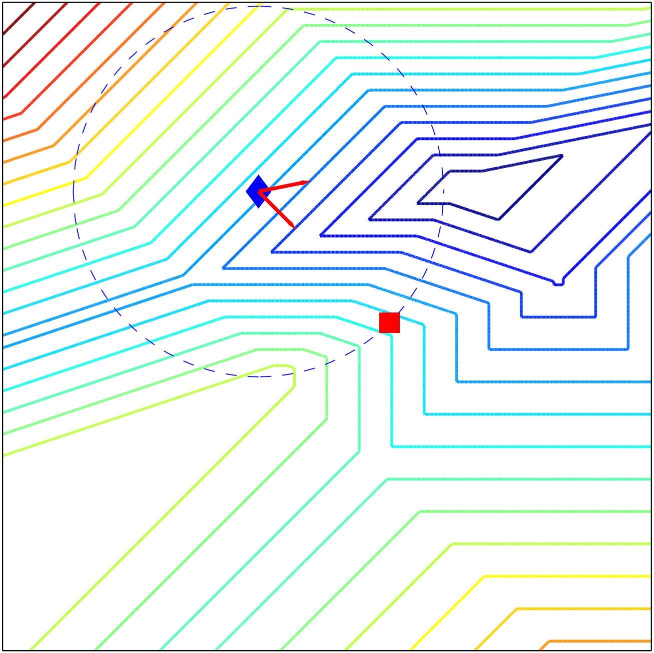


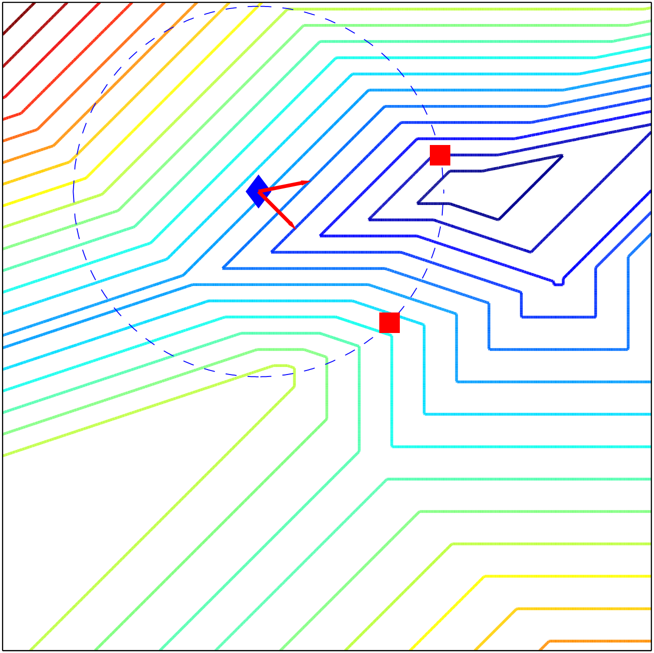


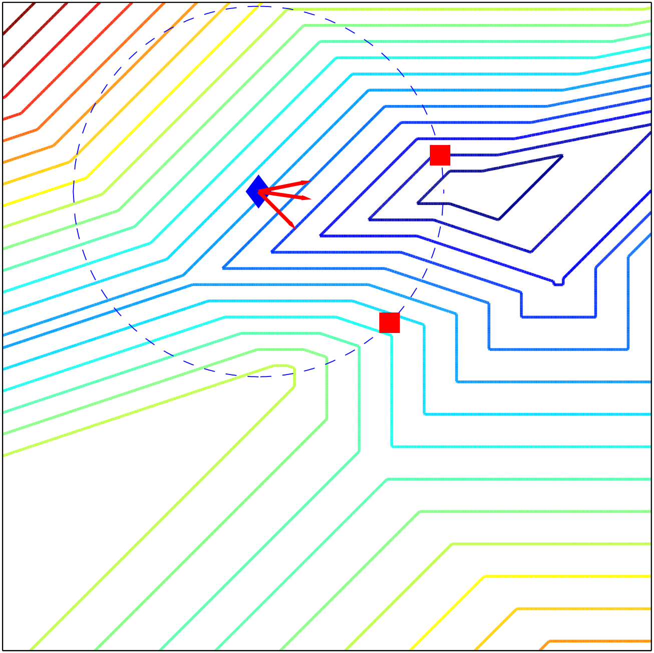


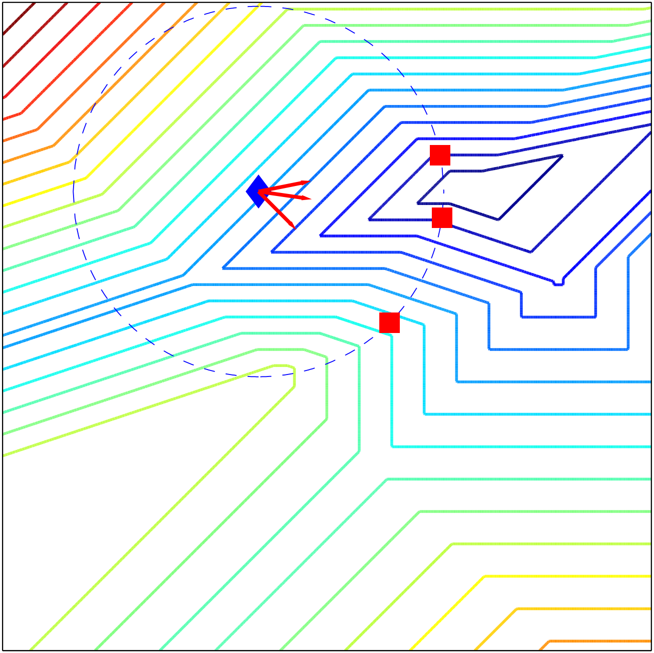


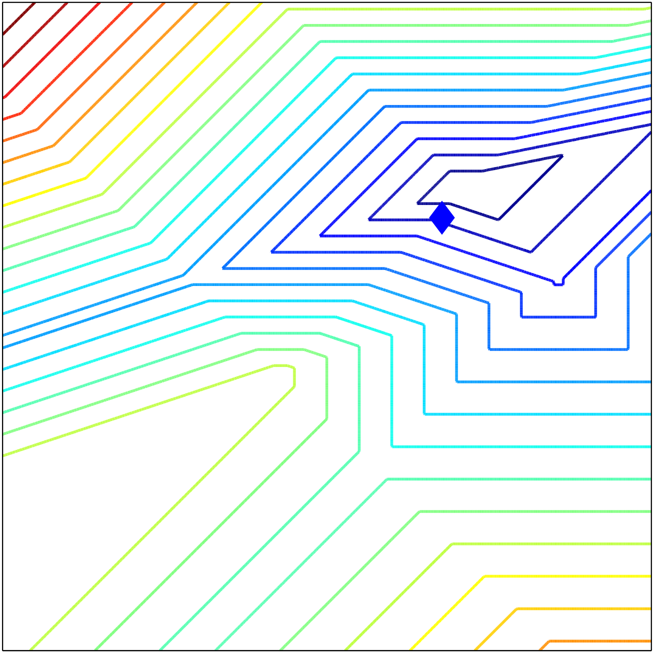


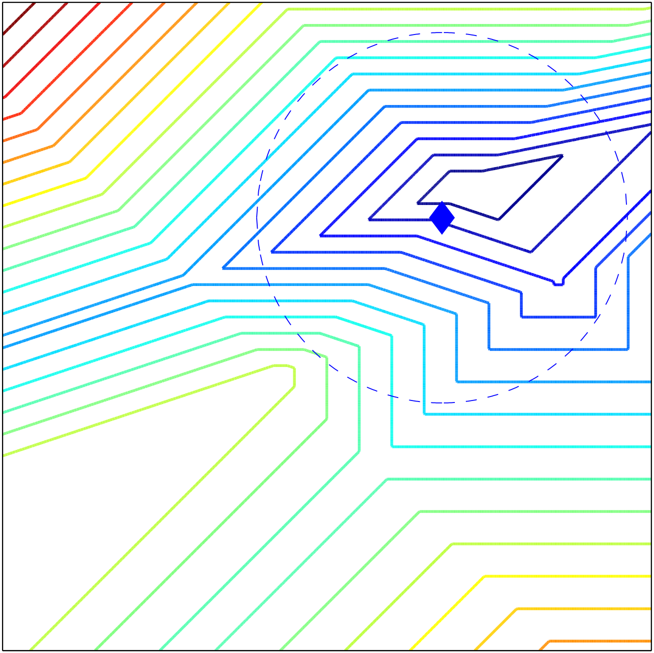


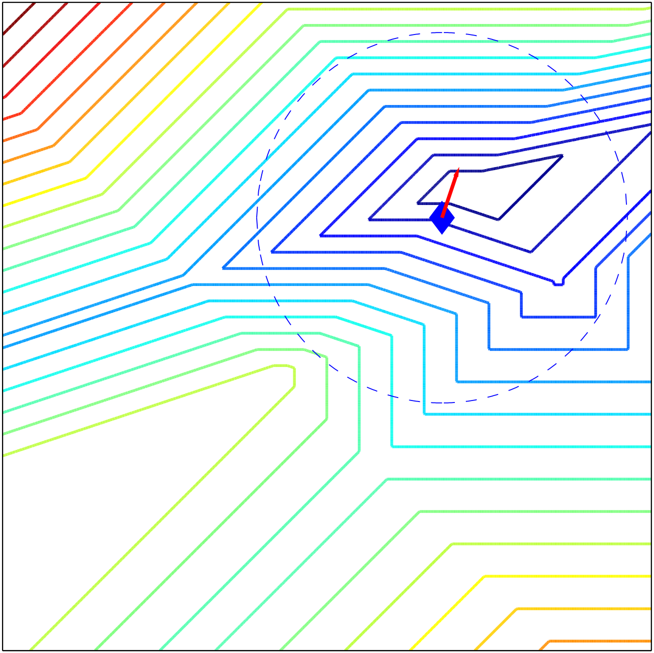


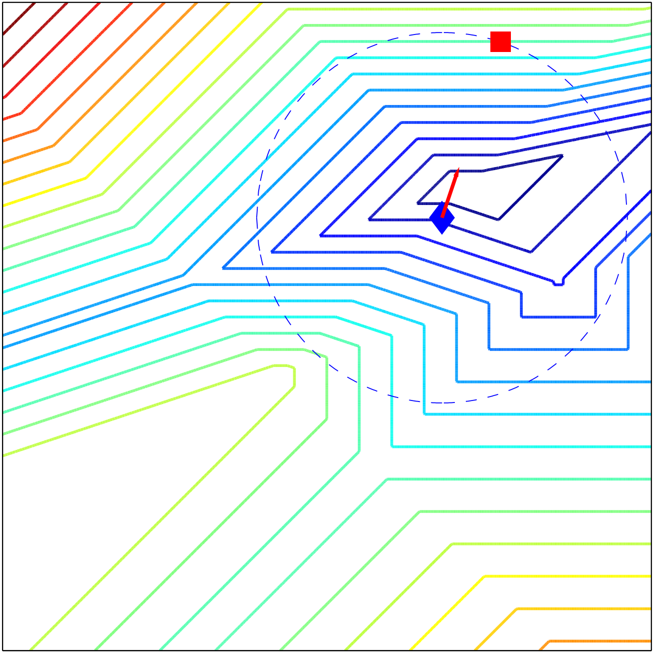


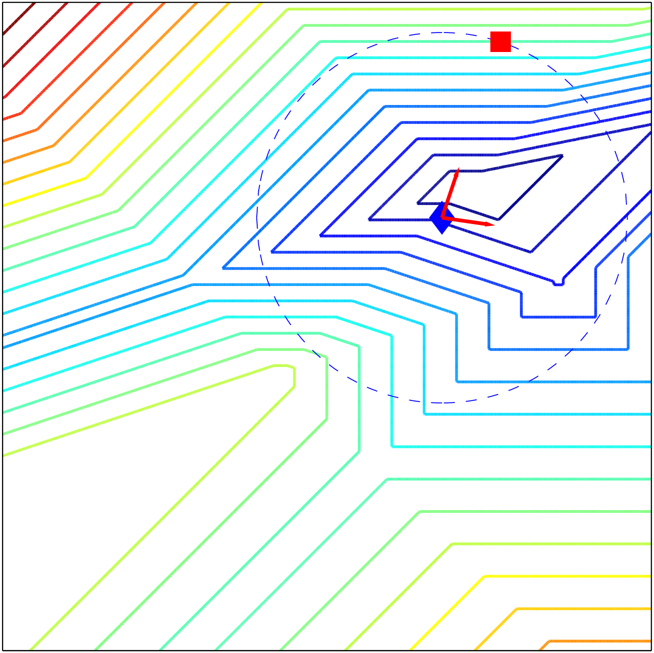


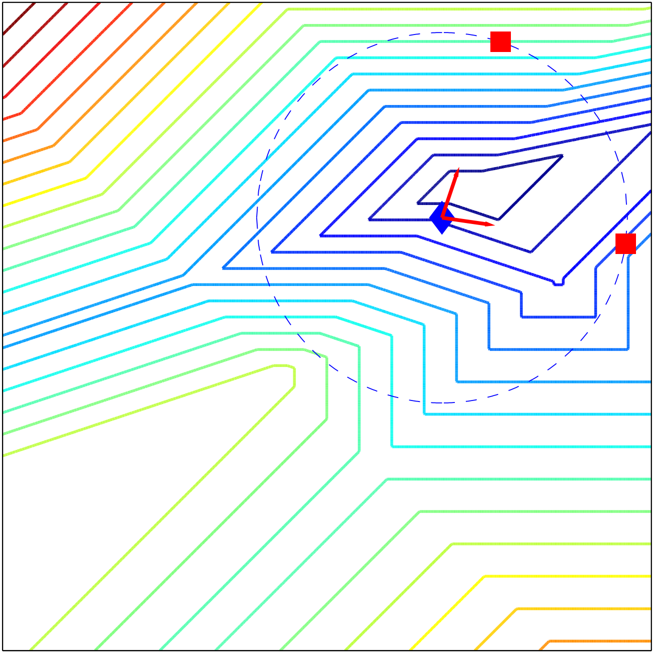


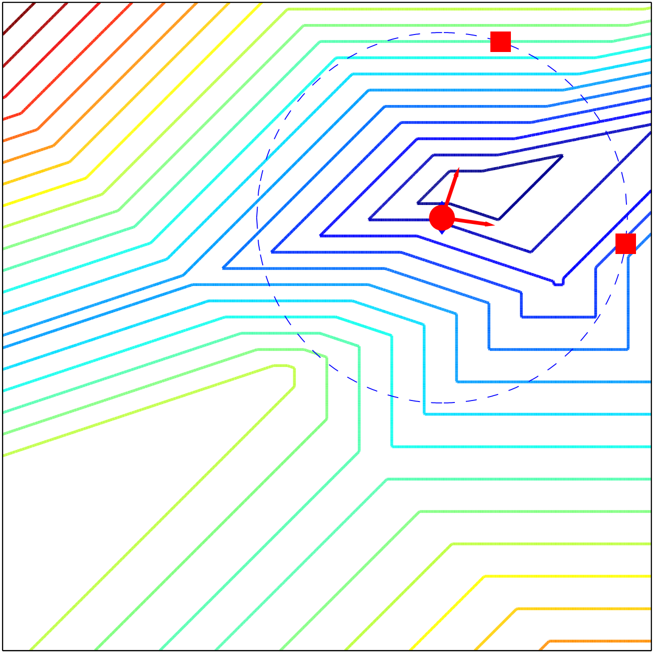


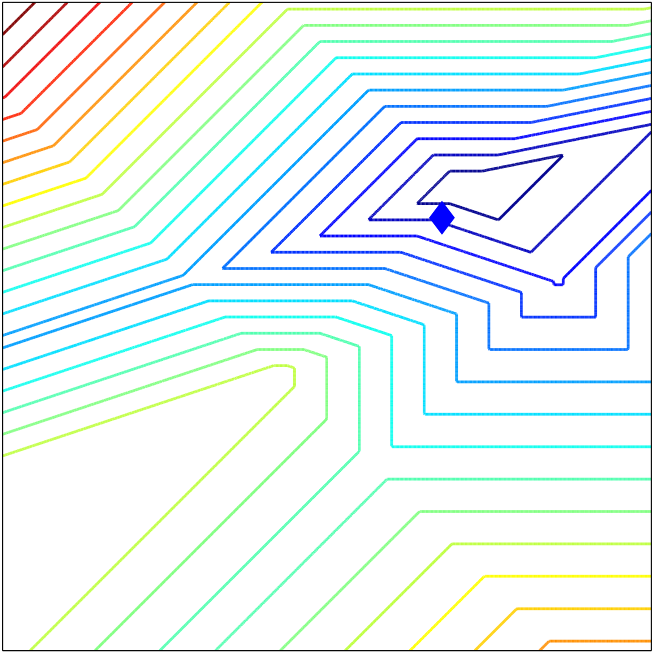


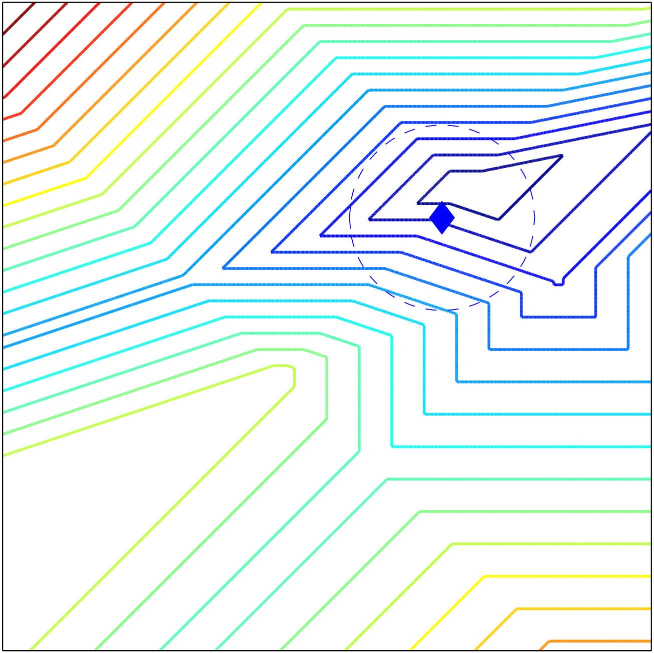


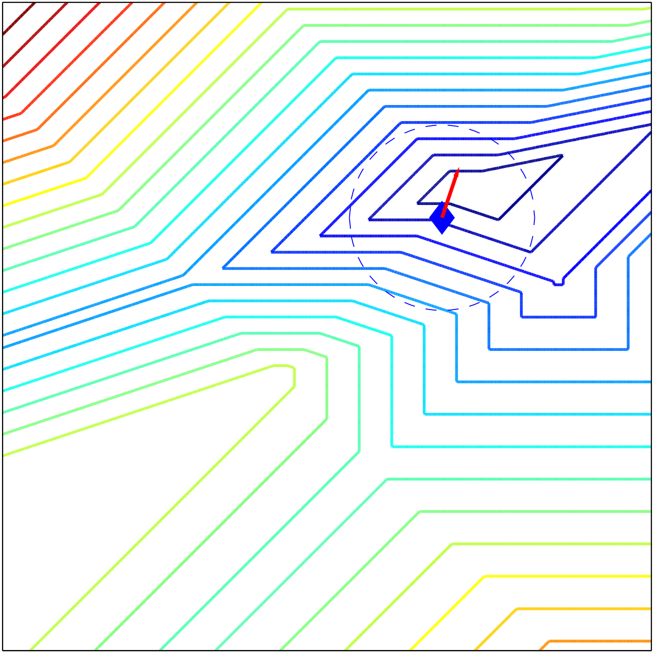


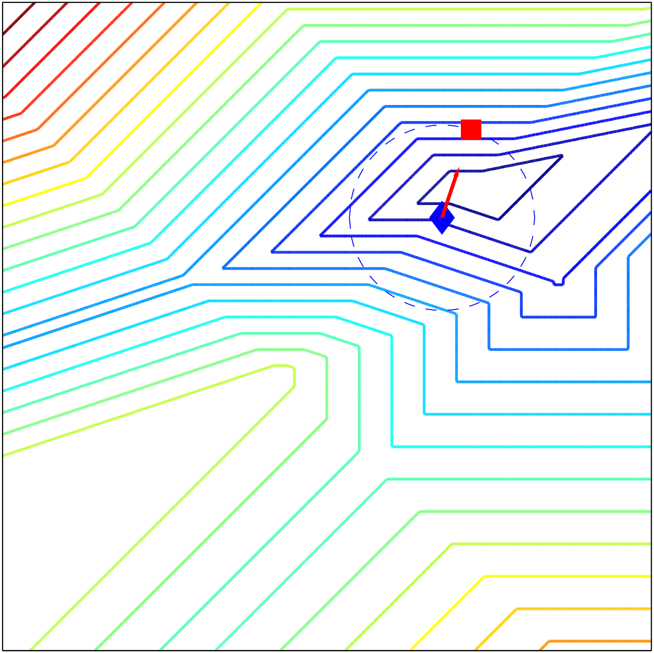


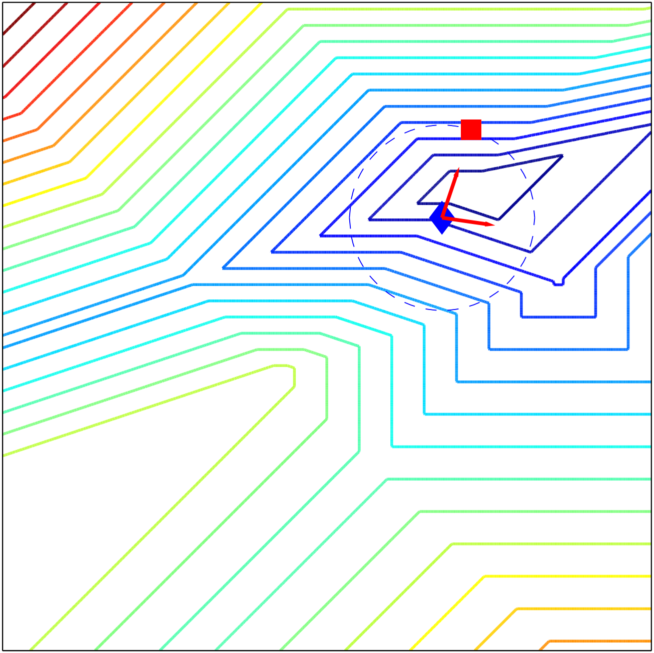


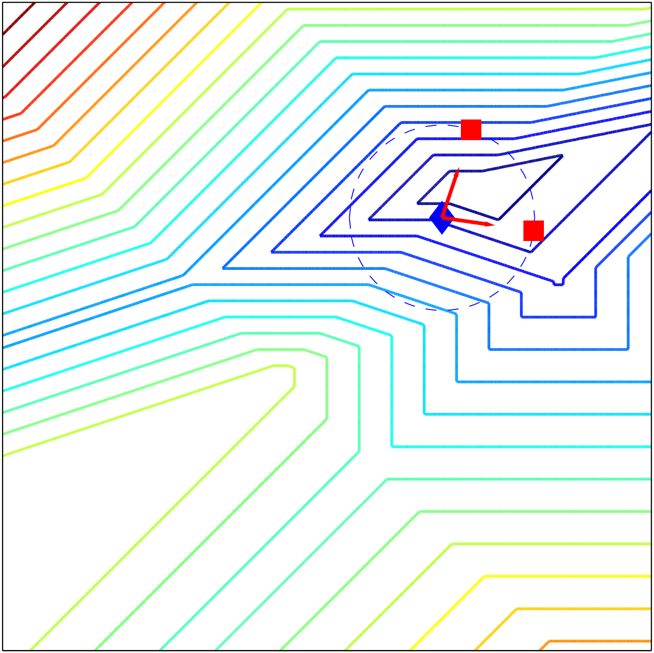


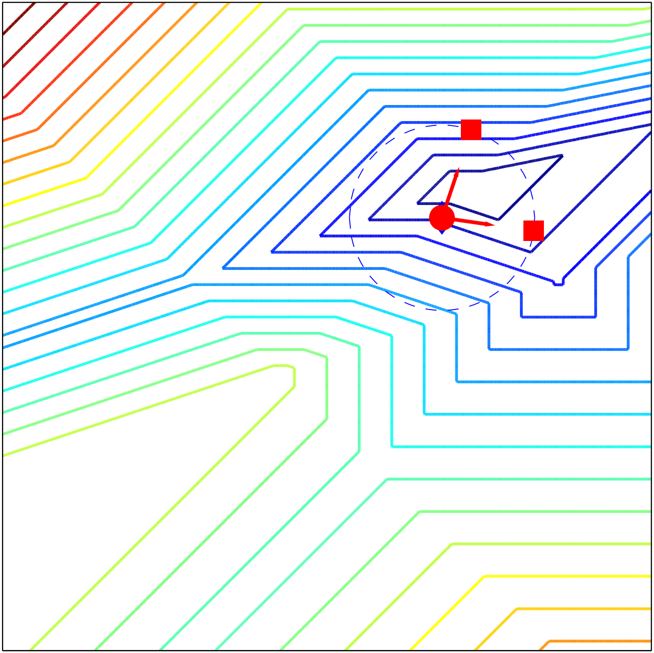


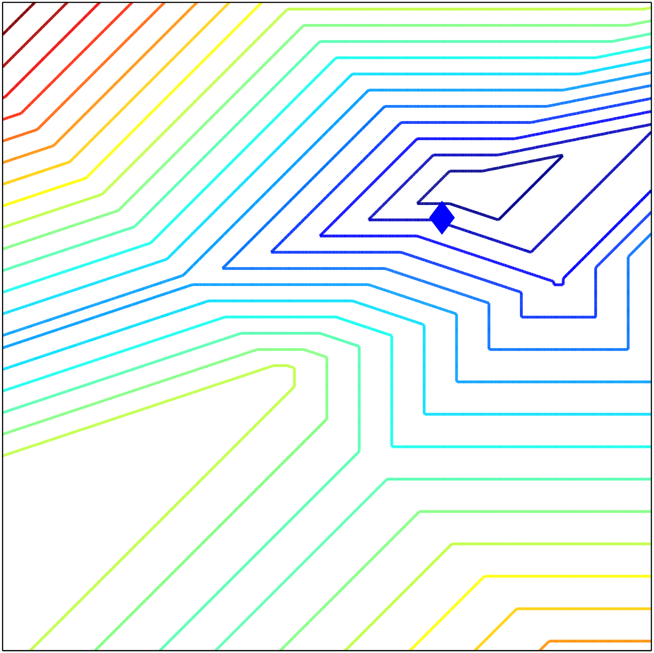


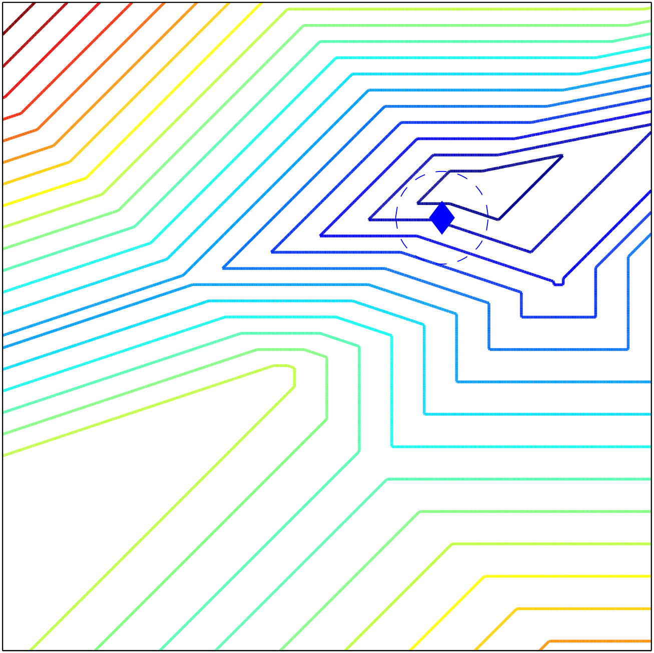


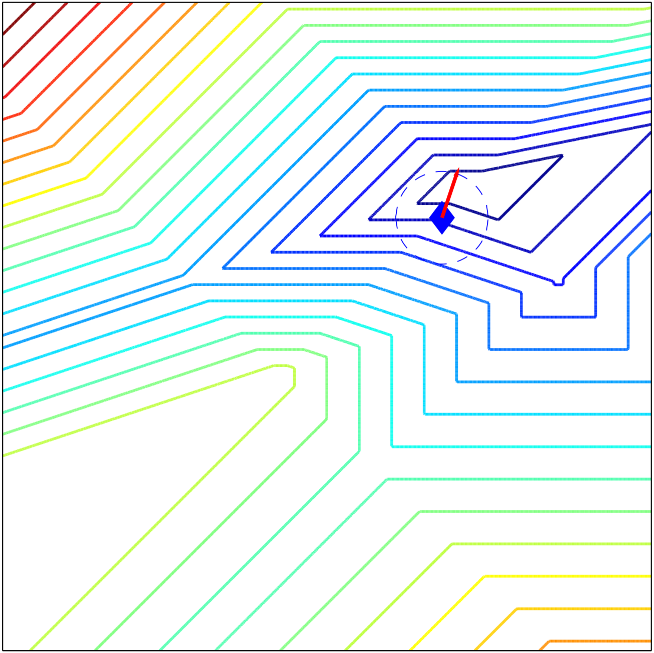


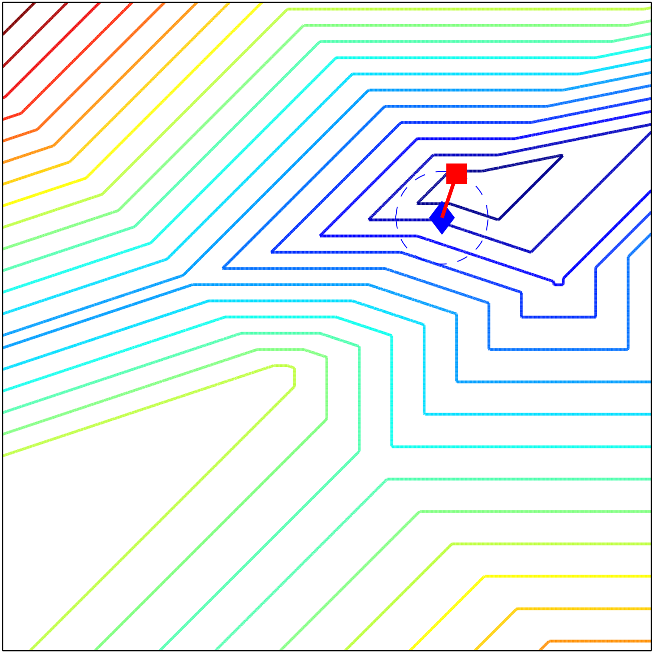


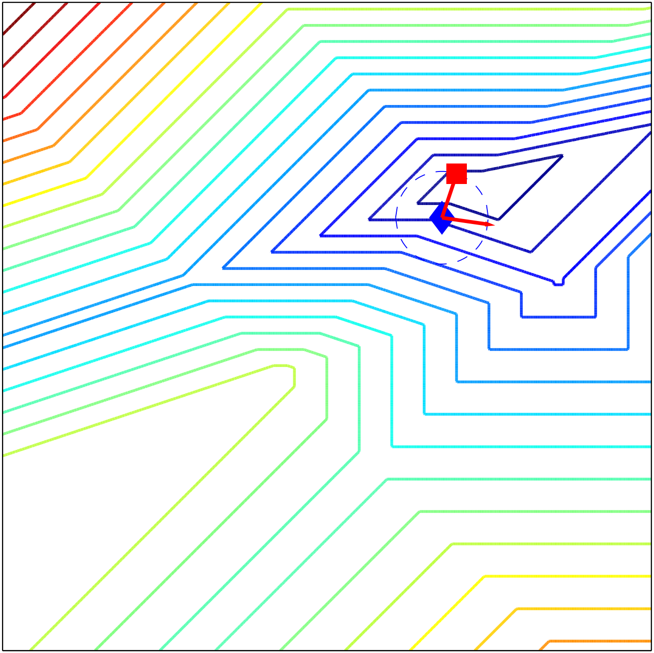


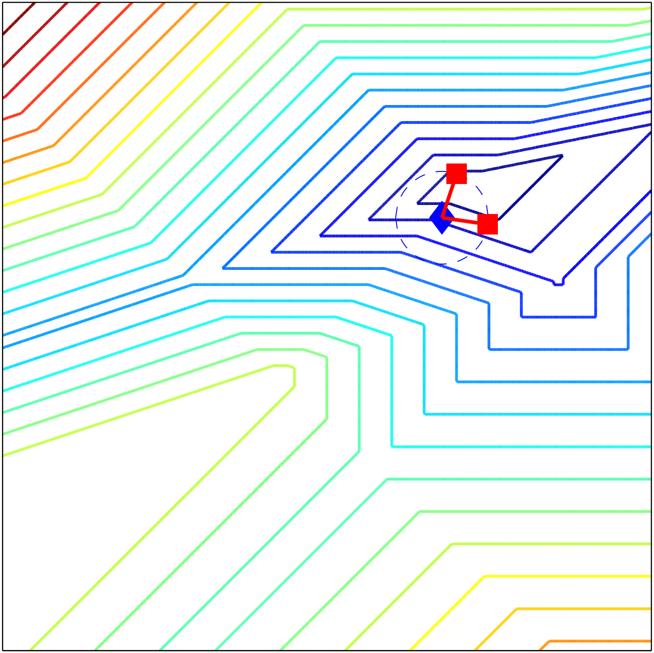


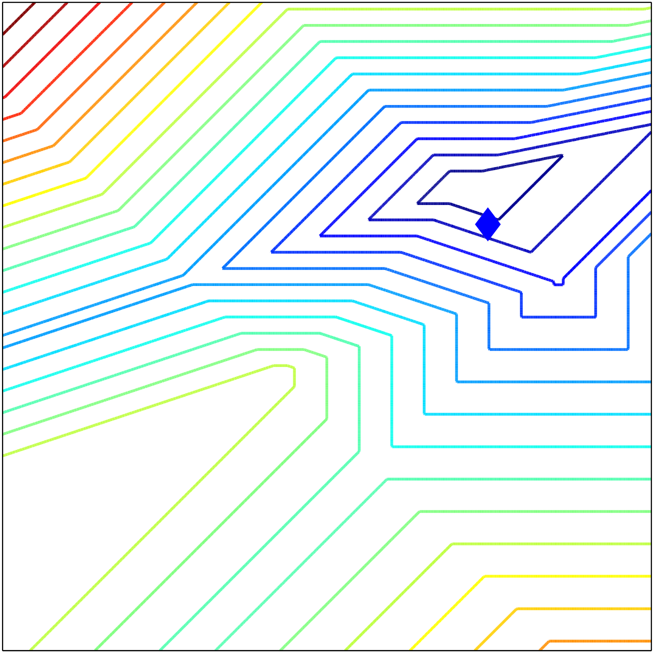


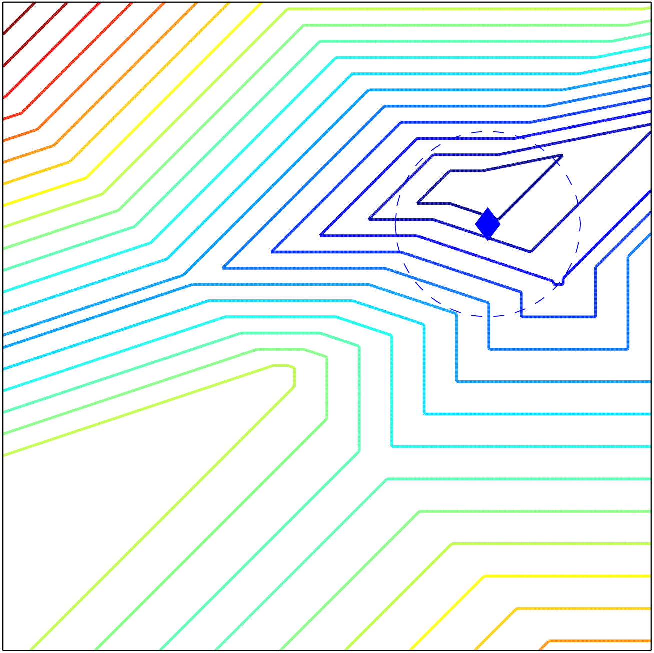


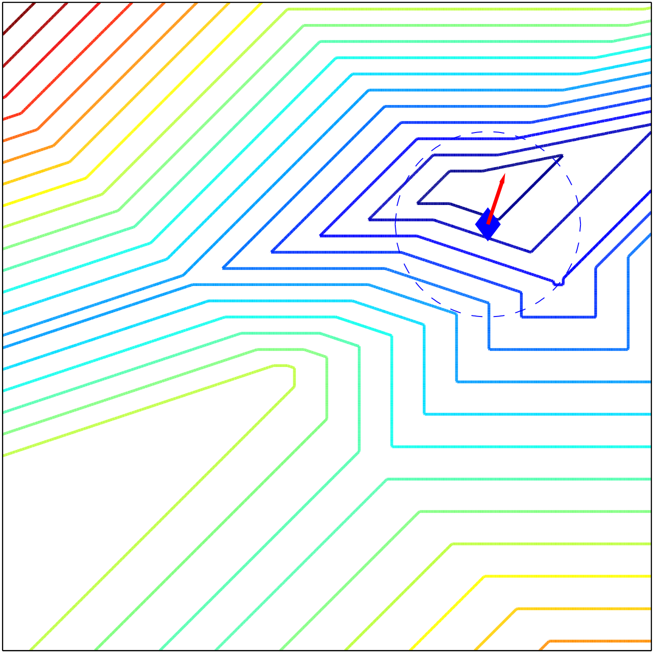


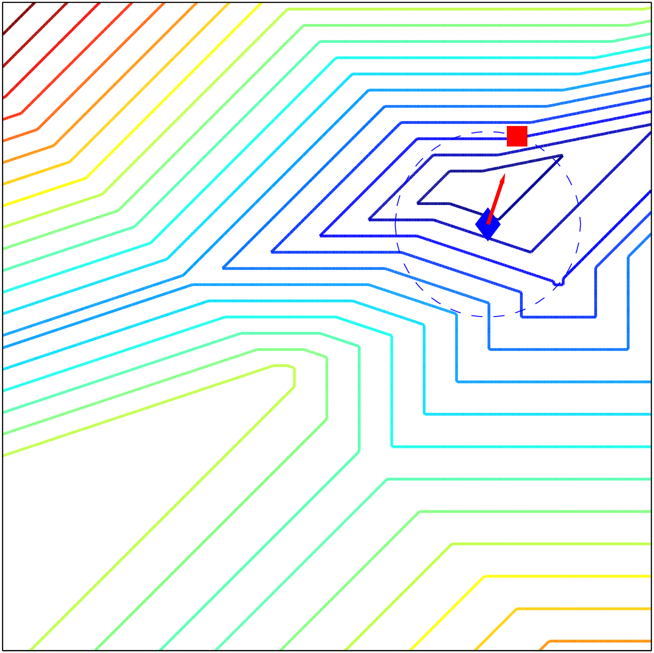


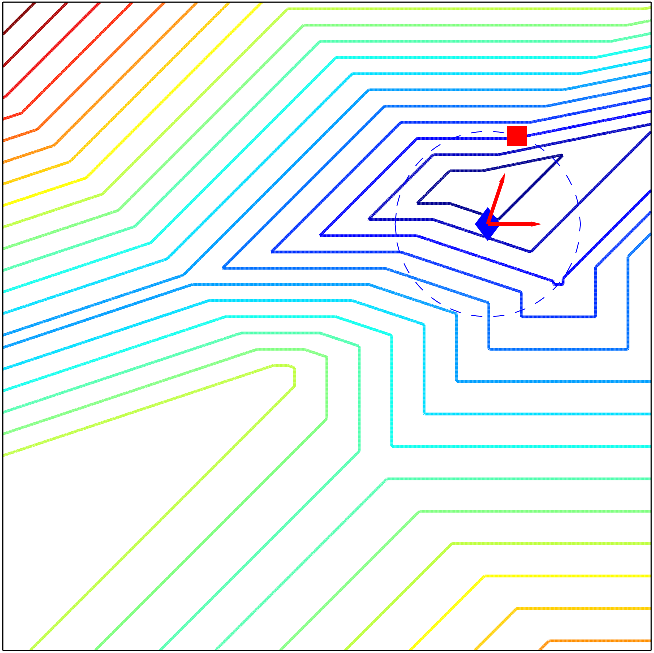


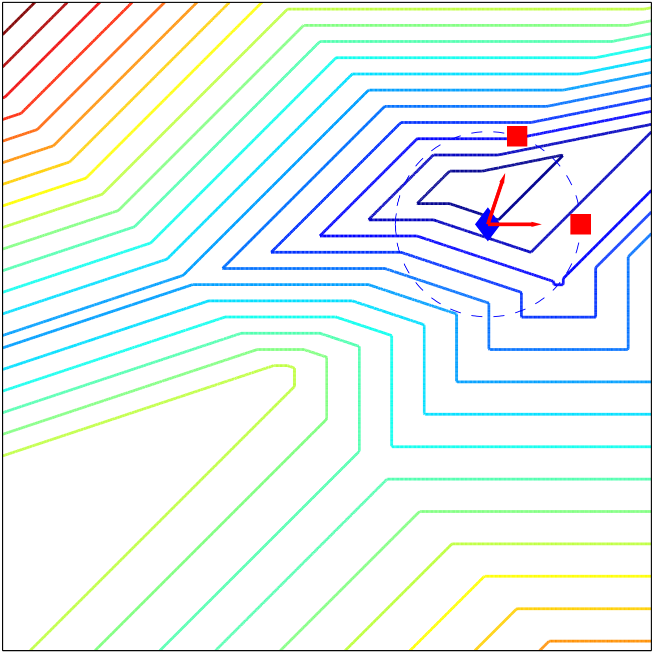


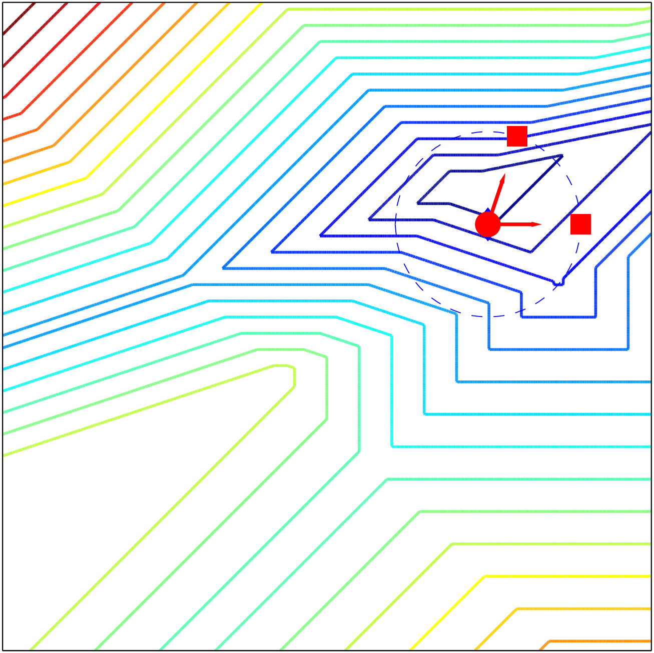


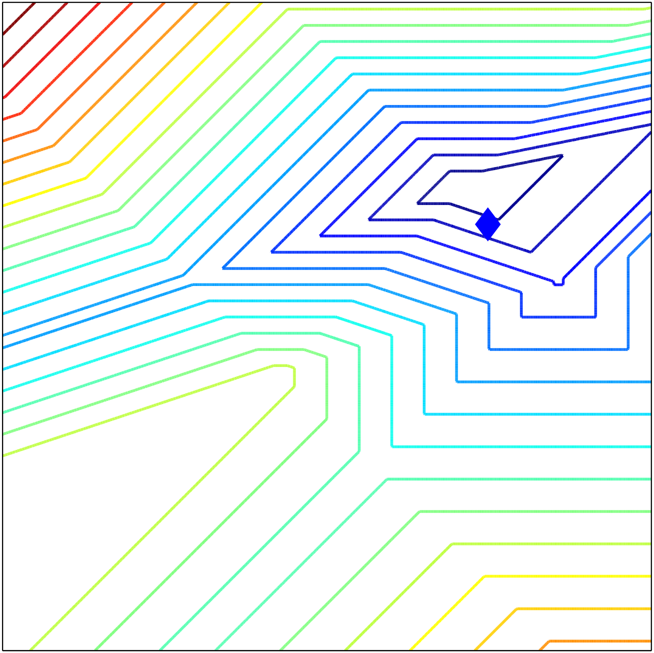


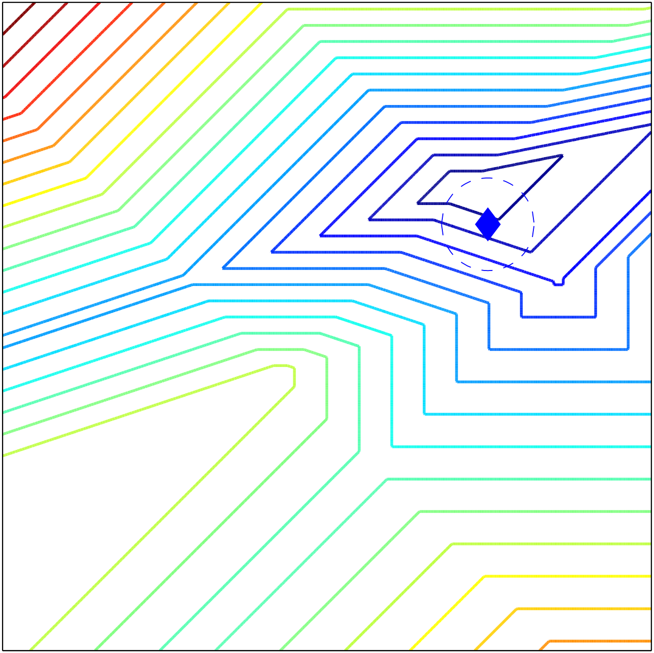


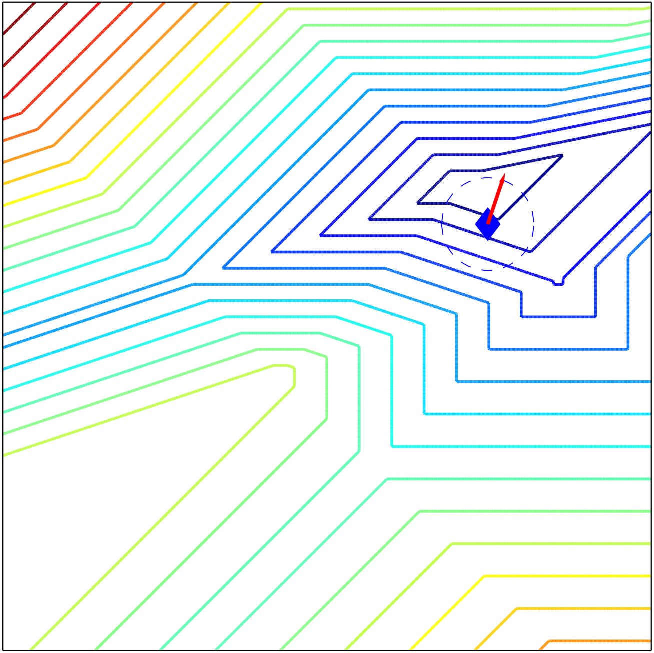


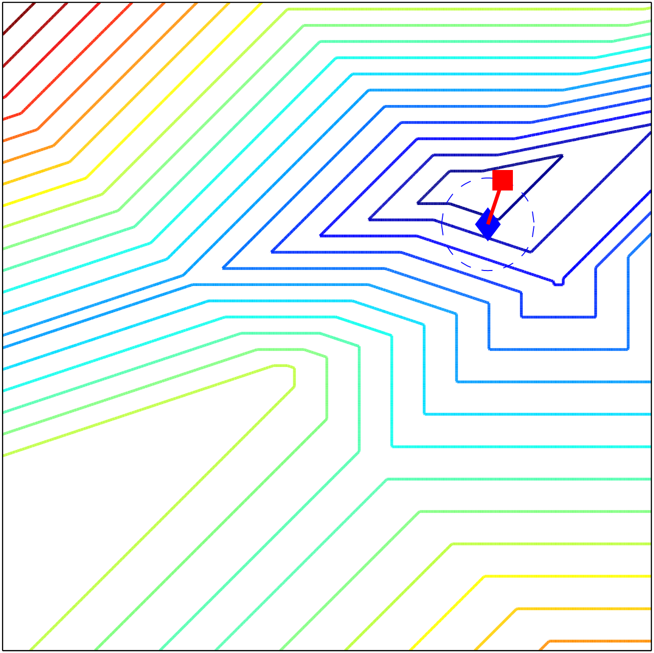


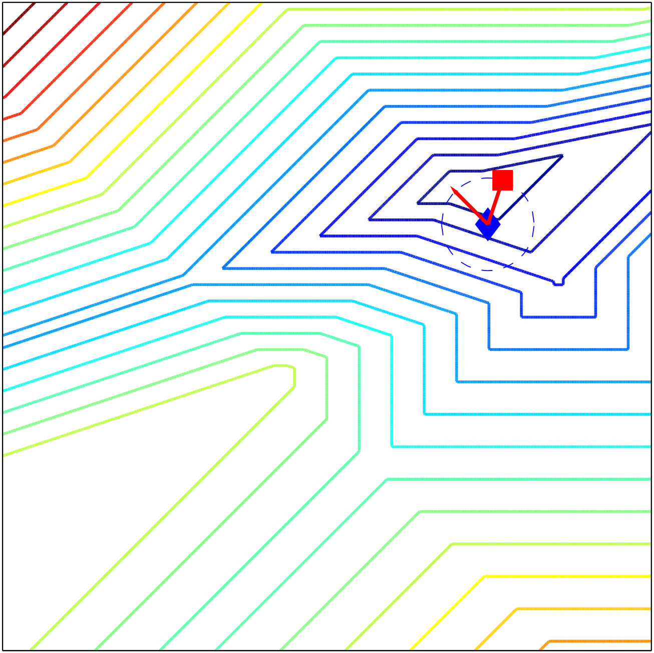


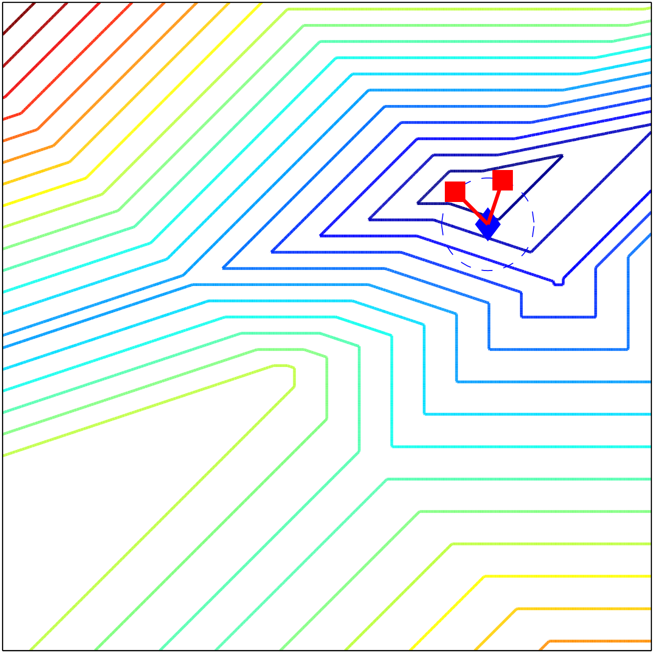


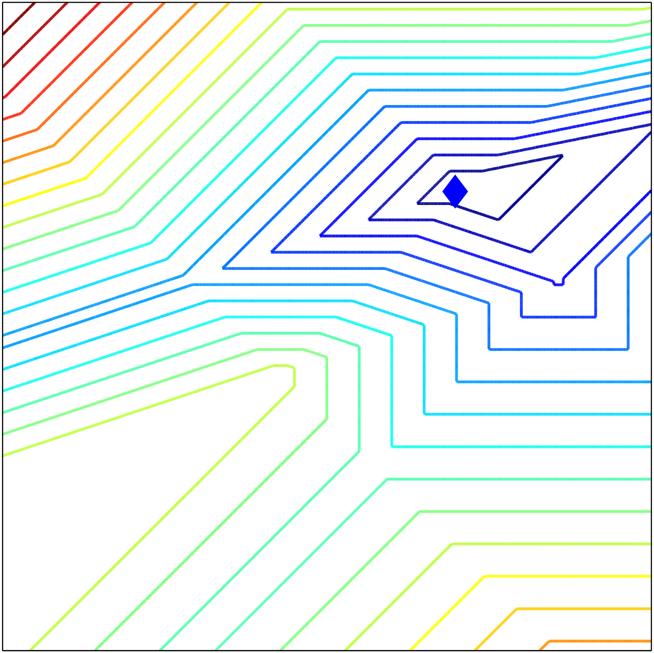












Generator set

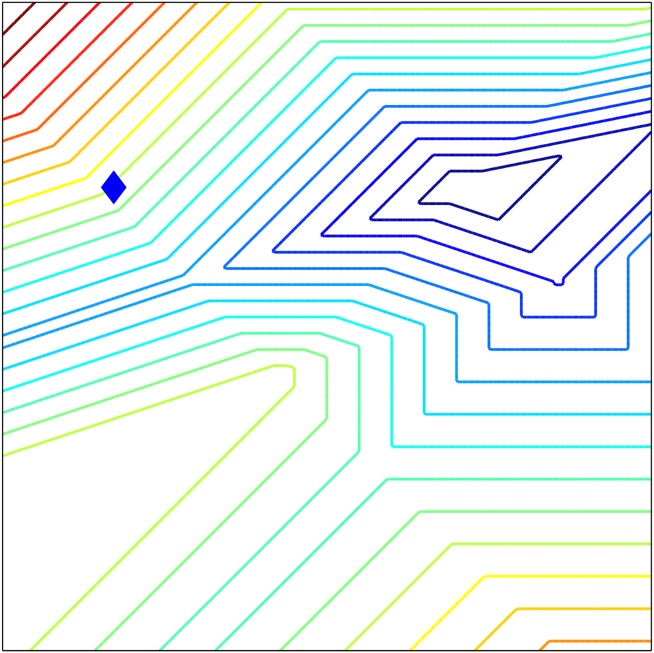
At some iterate x^k,

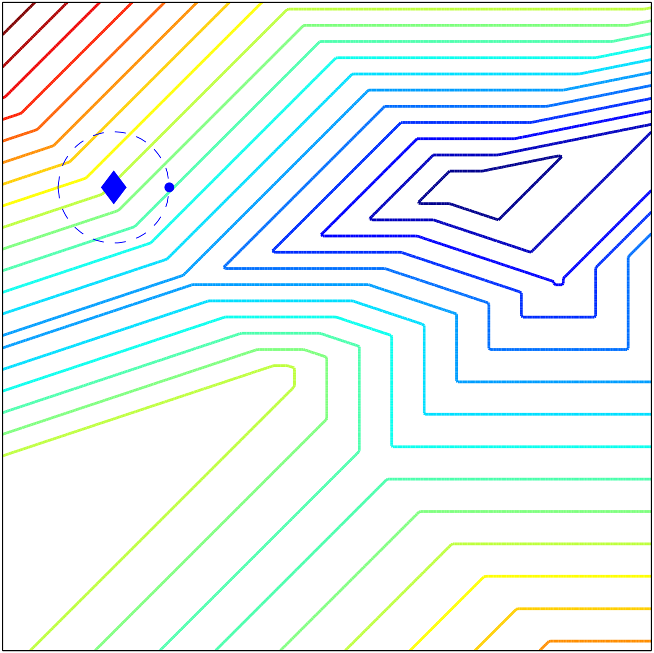
$$\mathfrak{G}^k \triangleq \bigcup_{i \in I_h(F(x^k))} \left\{ \nabla\psi(x^k) + \nabla M(x^k) a_i \right\}$$

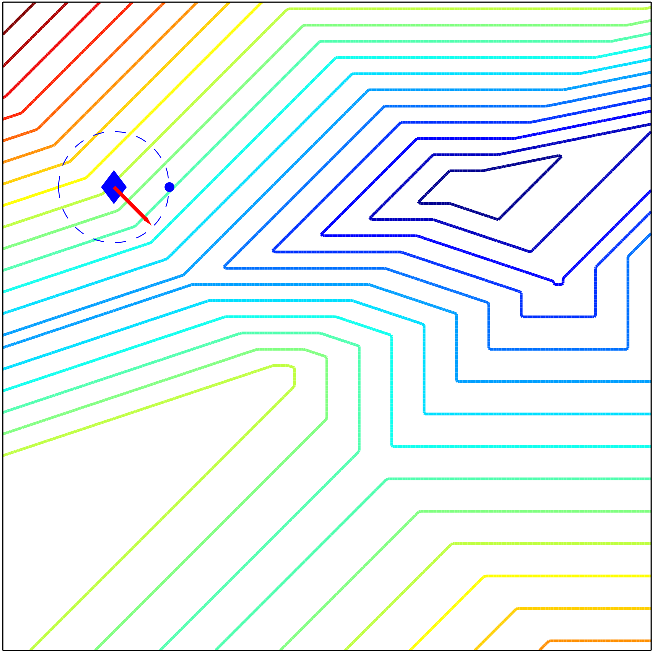
where $I_h(F(x^k))$ is the set of essentially active indices.

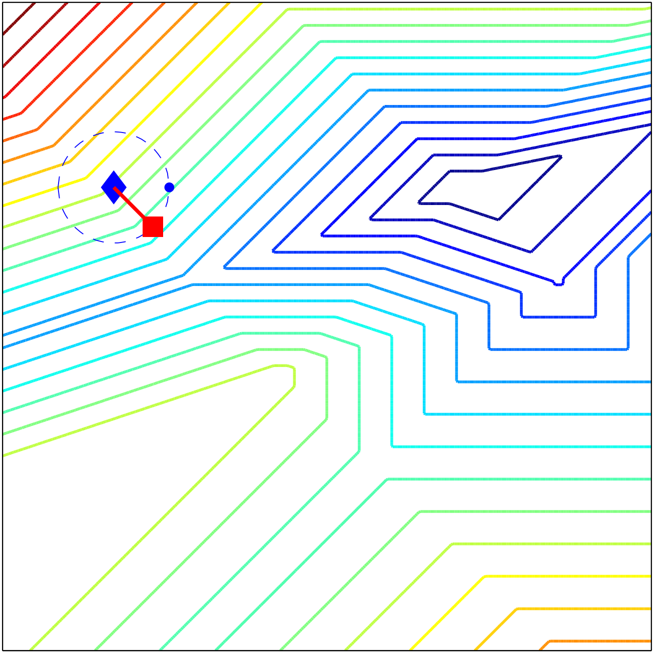
Or, given a set of points $Y = \left\{x^k, y^2, \ldots, y^p\right\} \subset \mathcal{B}(x^k, \Delta_k)$,

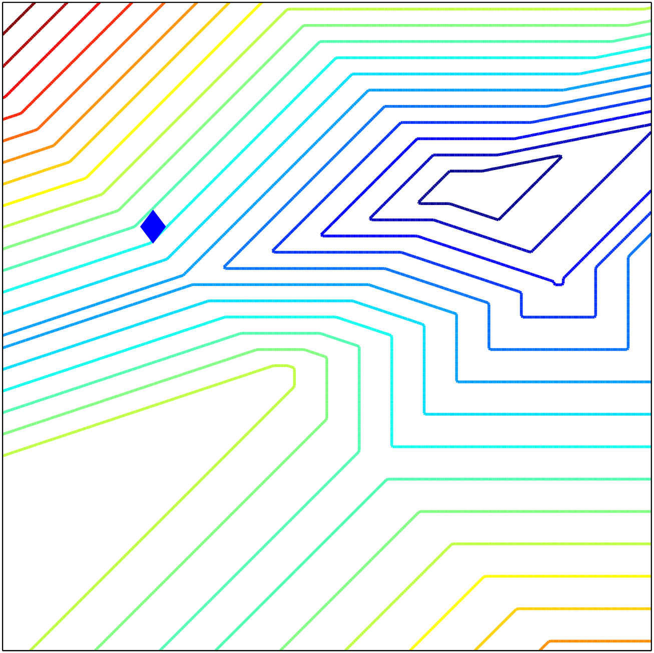
$$\mathfrak{G}^k \triangleq \bigcup_{y \in Y} \bigcup_{i \in I_h(F(y))} \left\{ \nabla\psi(x^k) + \nabla M(x^k) a_i \right\}$$

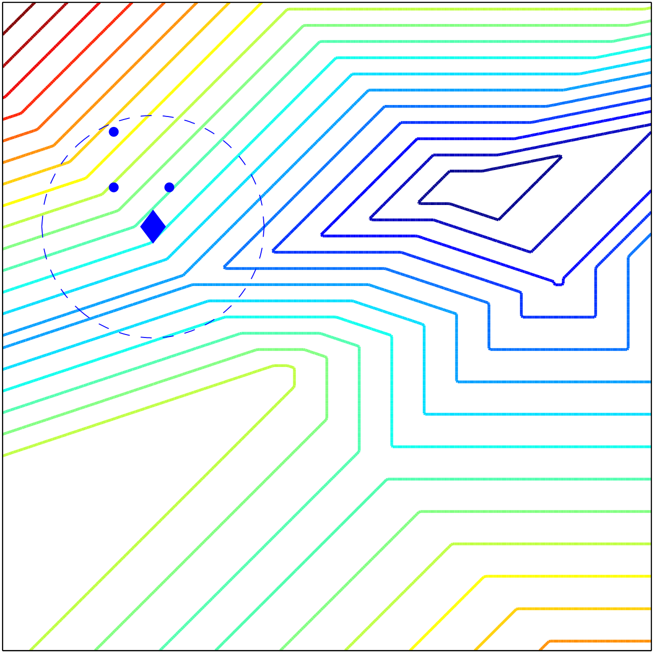


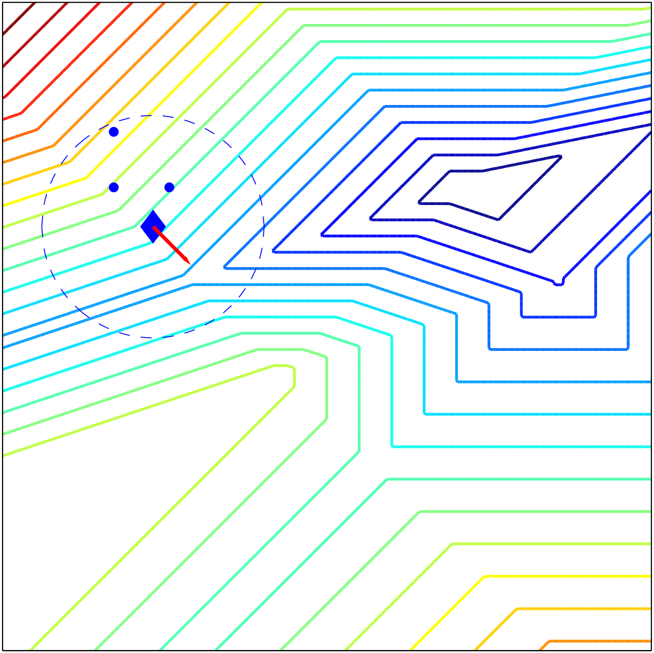


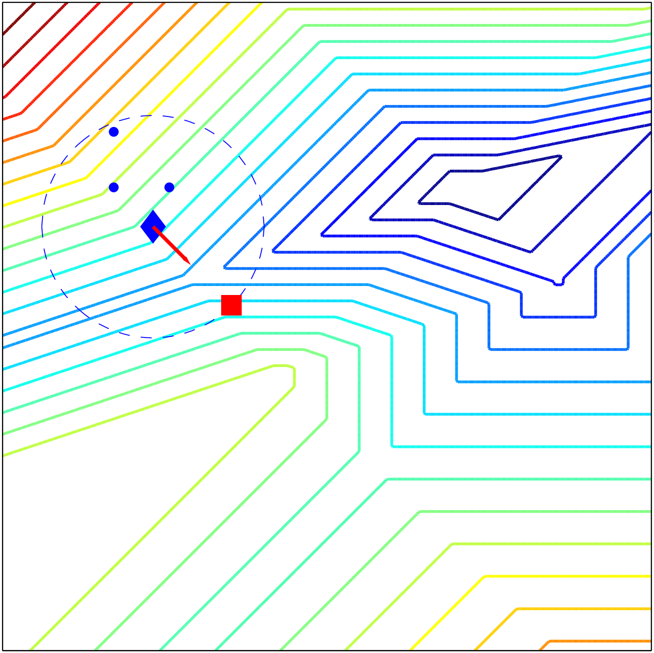


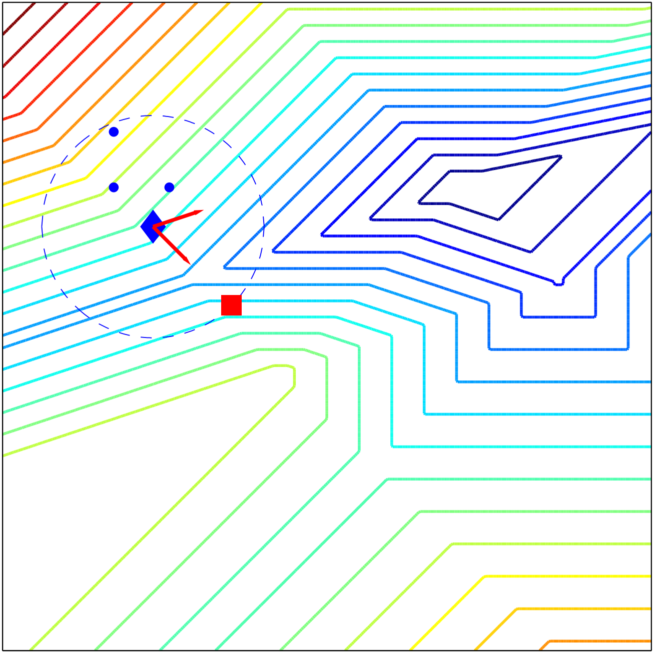


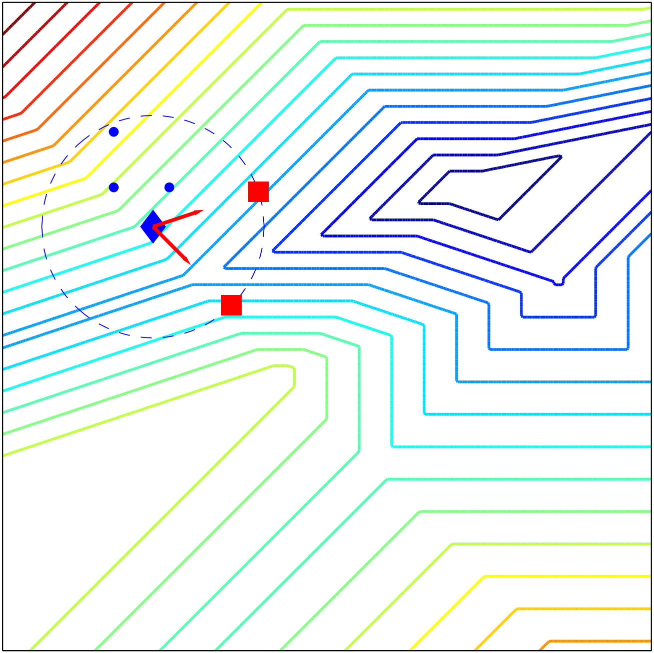


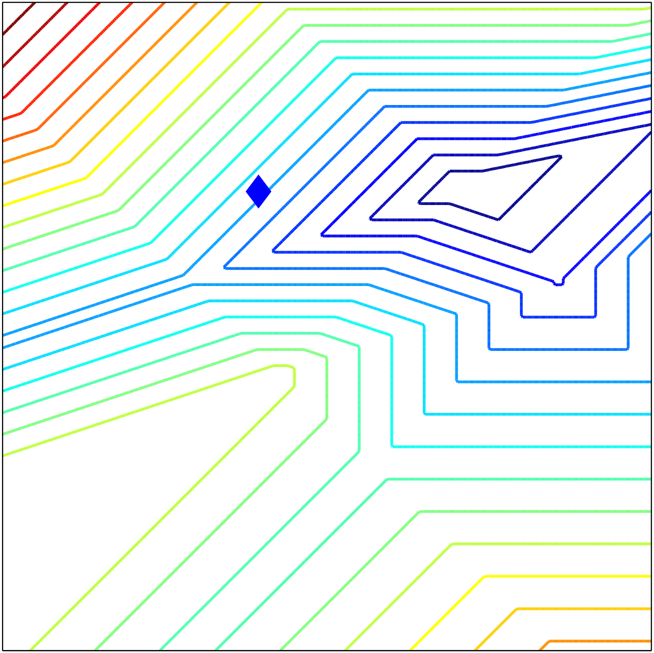


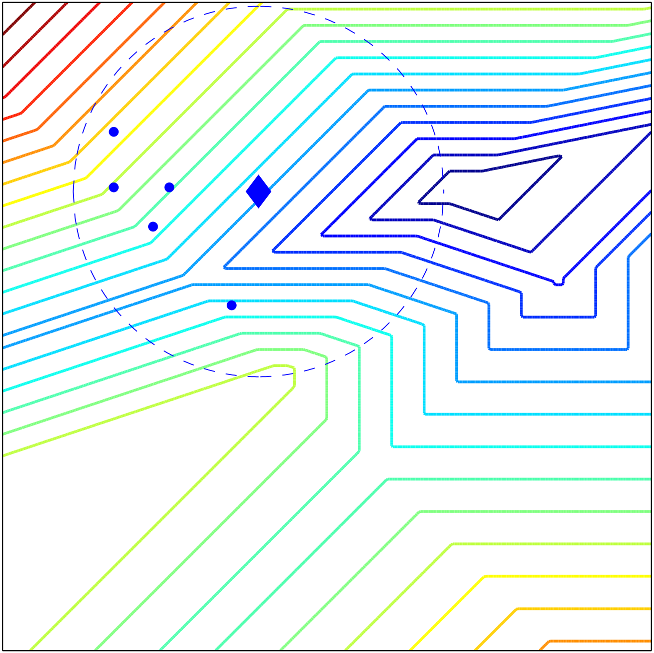


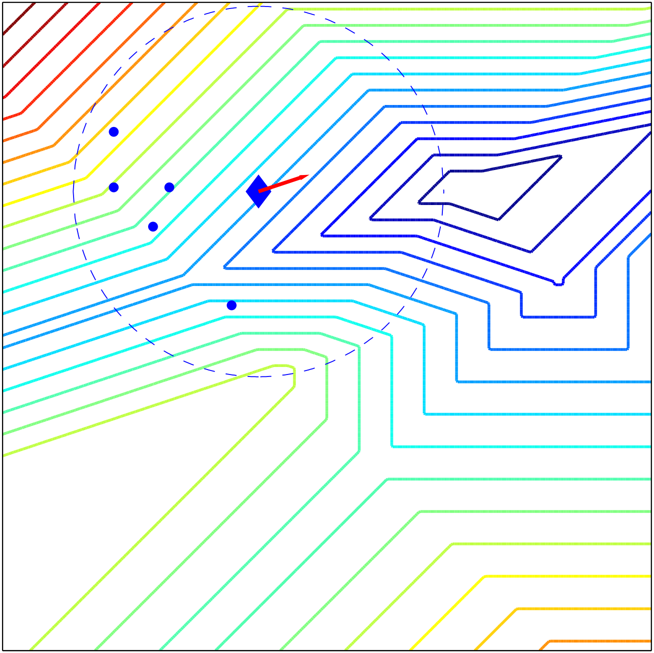


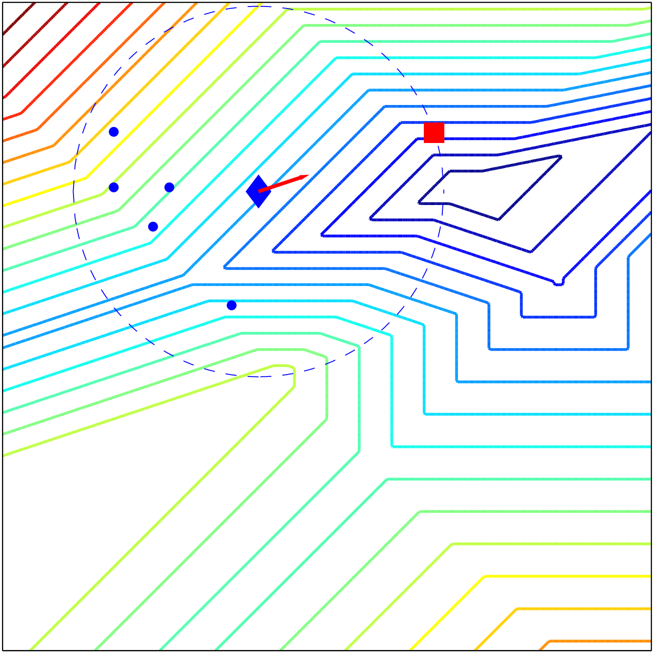


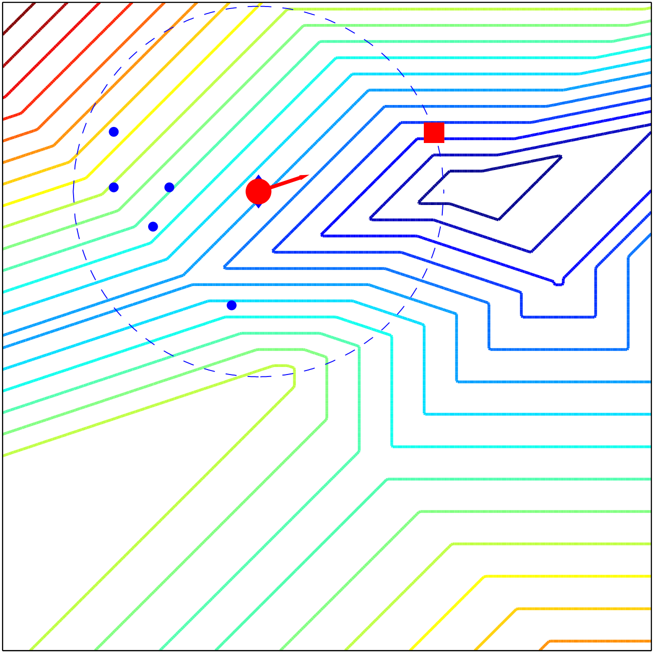


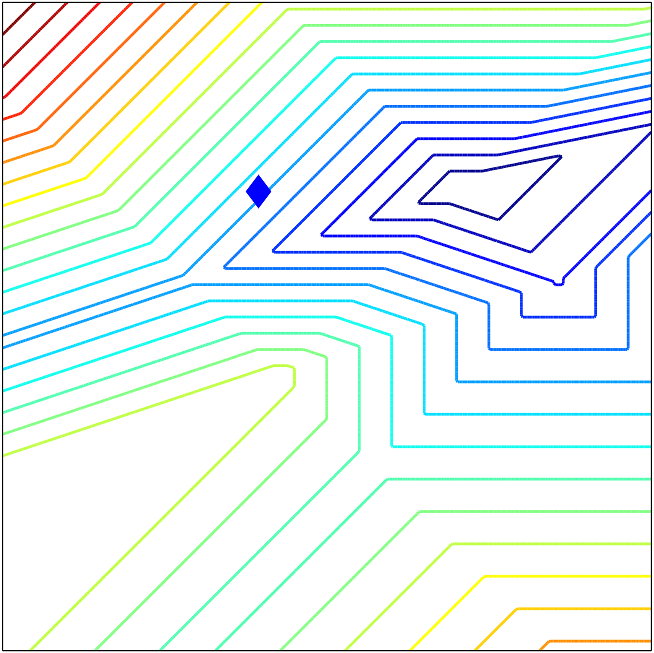


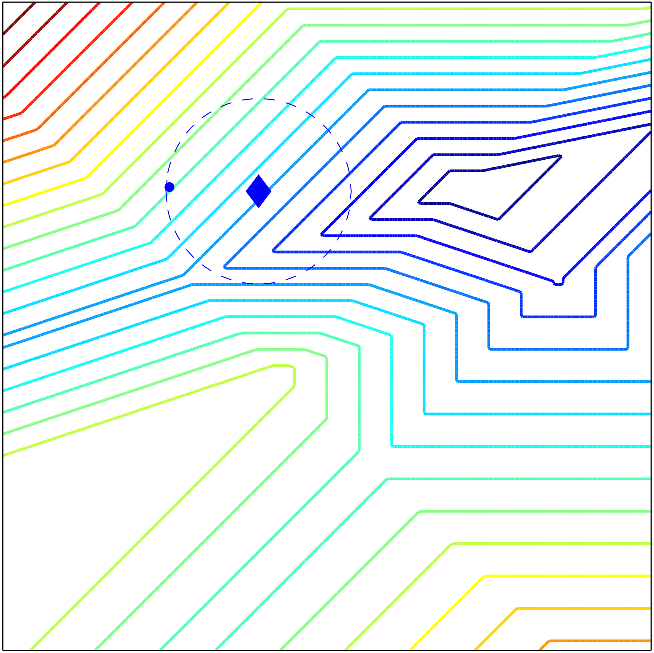


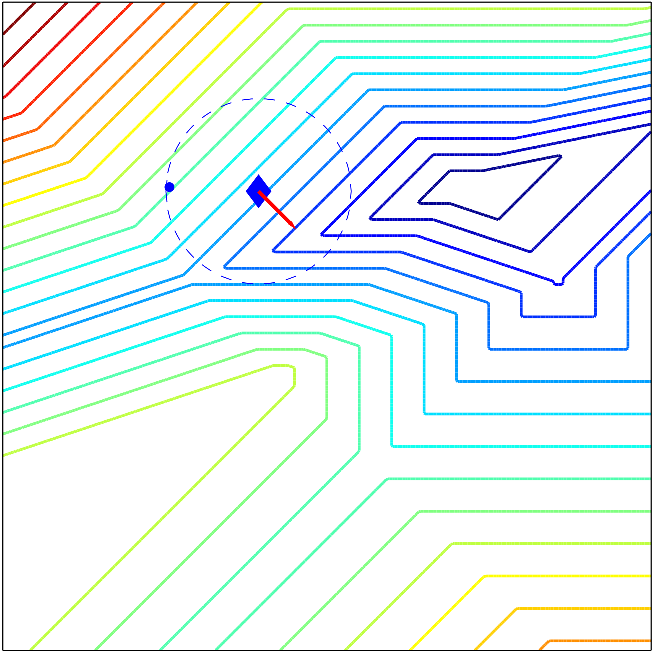


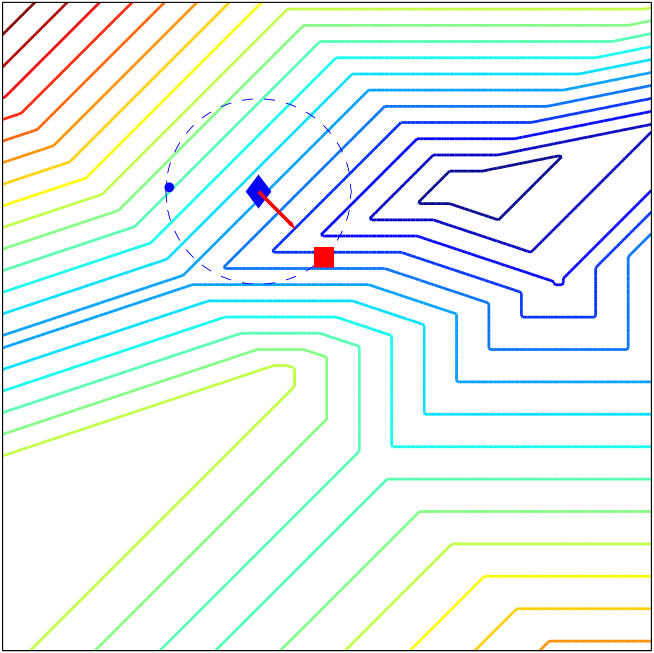


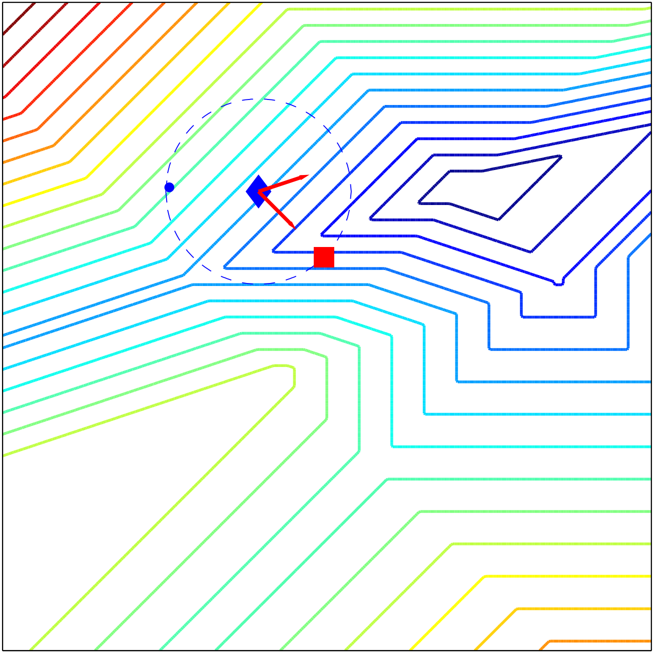


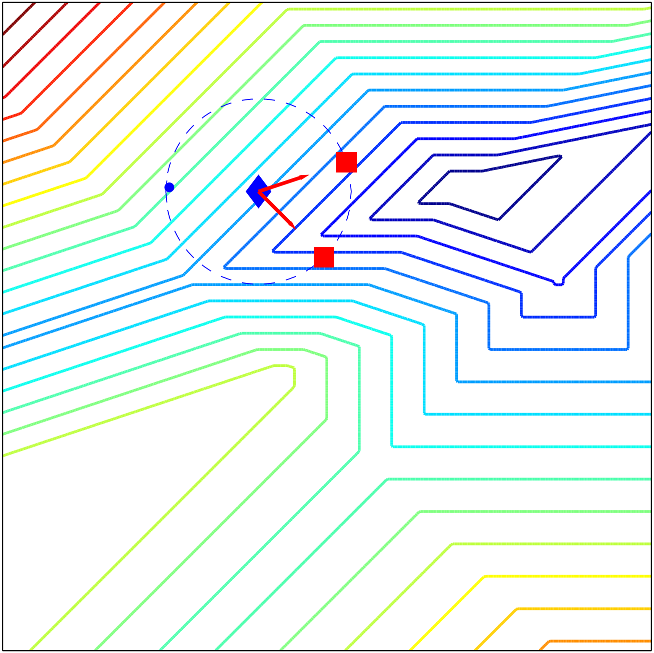


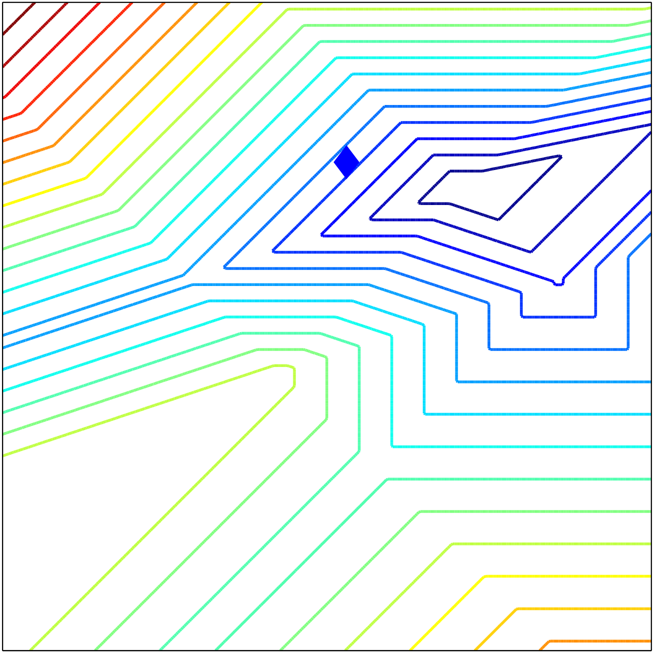


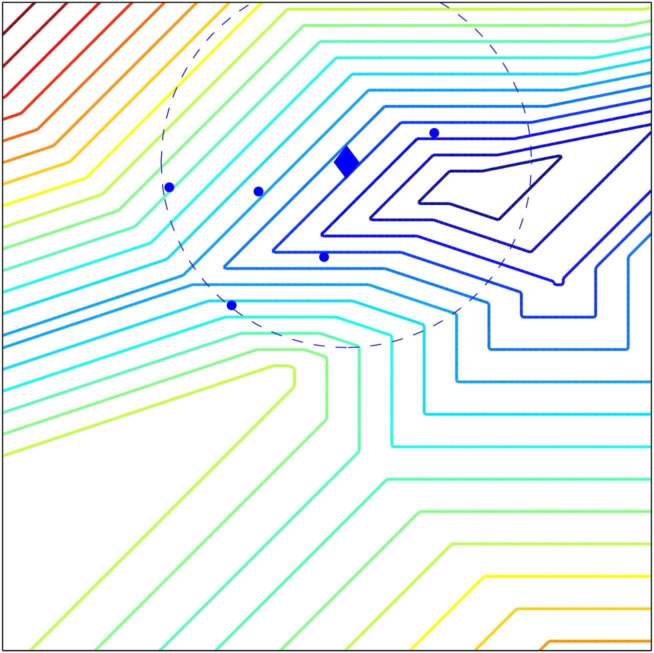


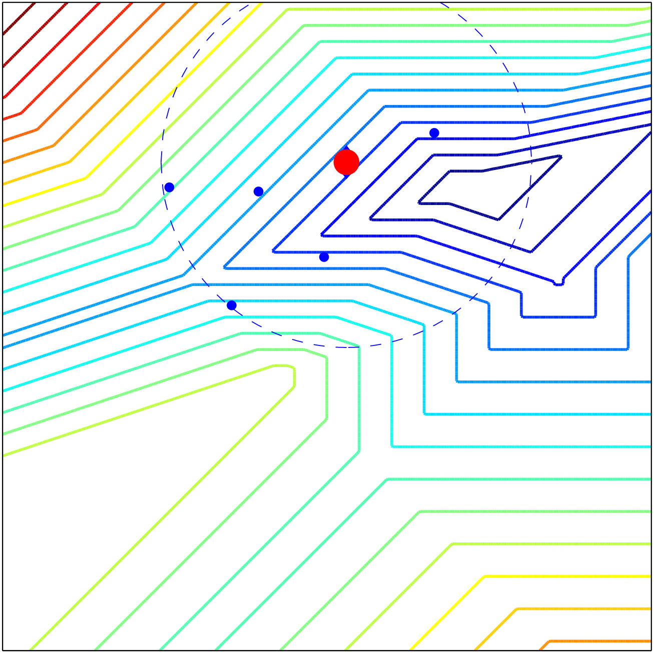


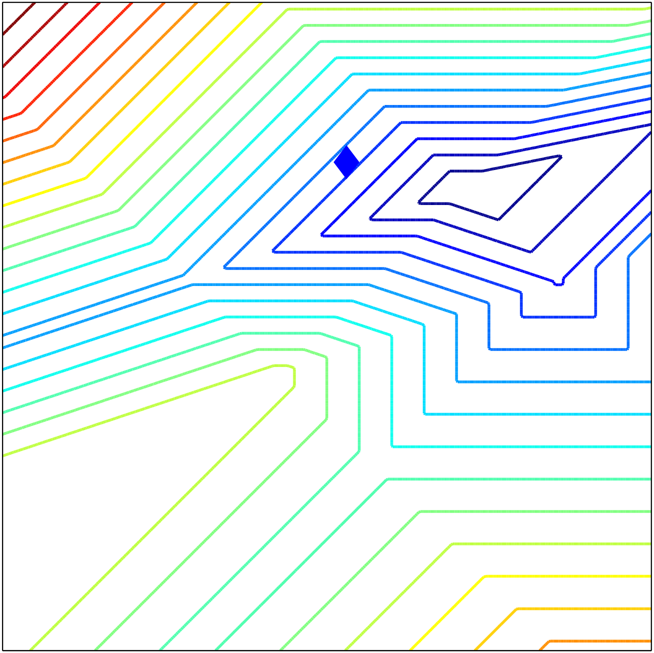


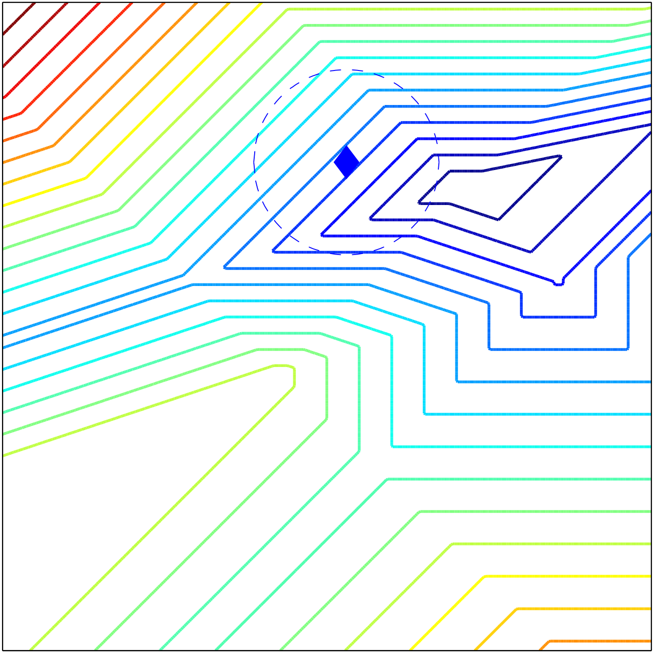


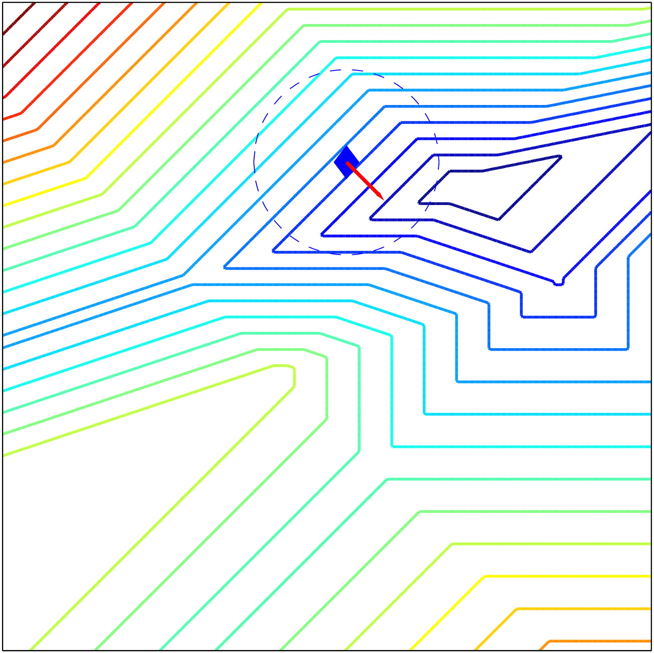


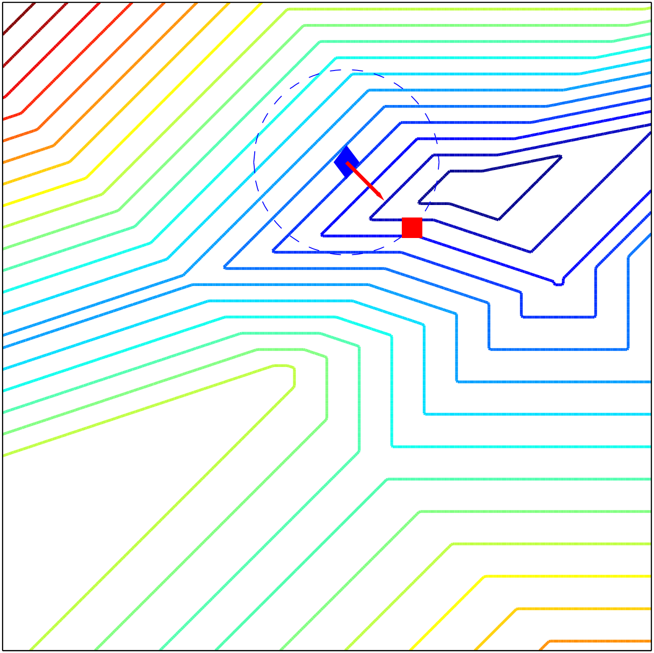


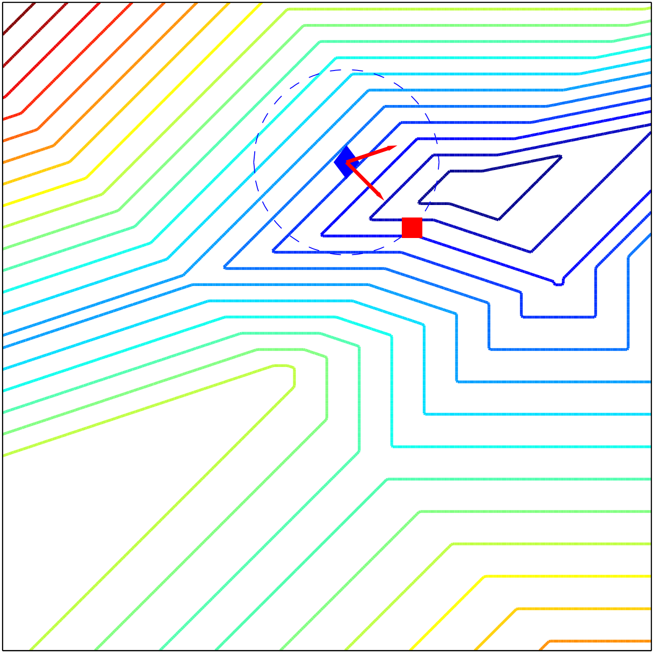


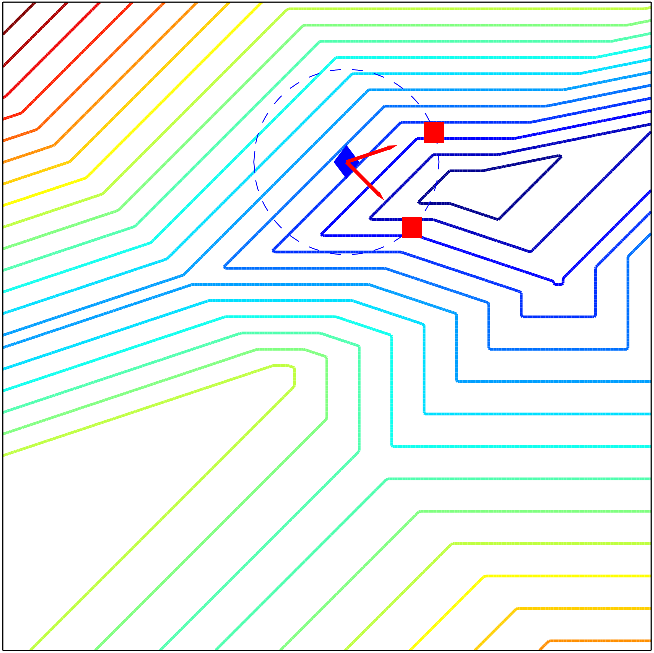


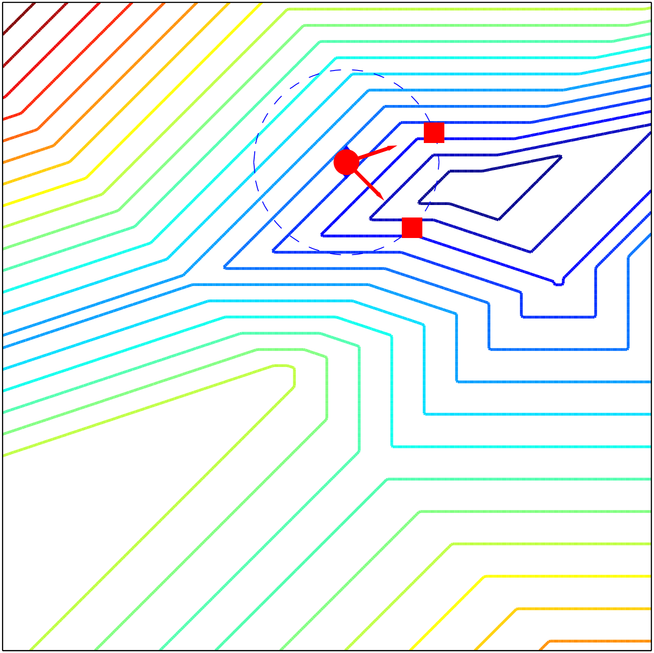


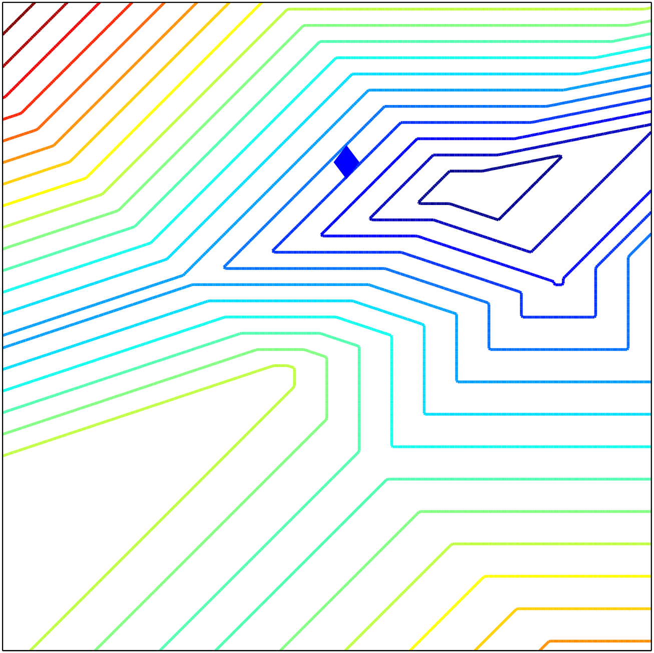


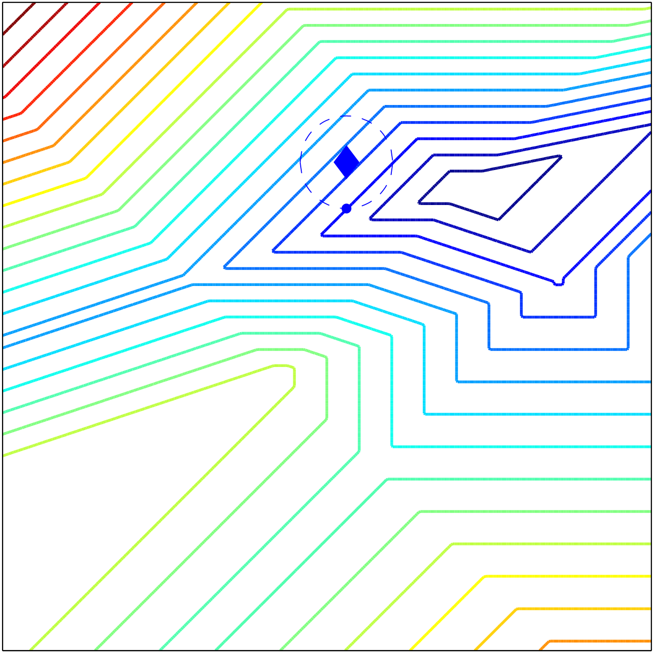


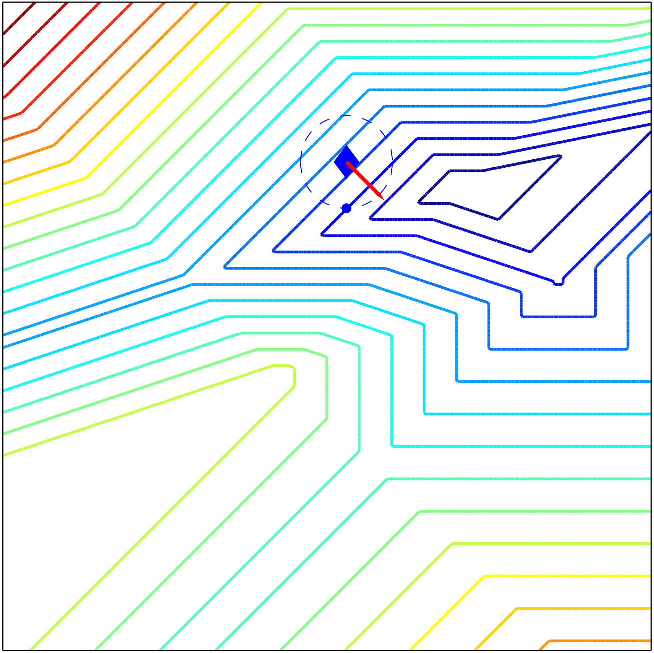


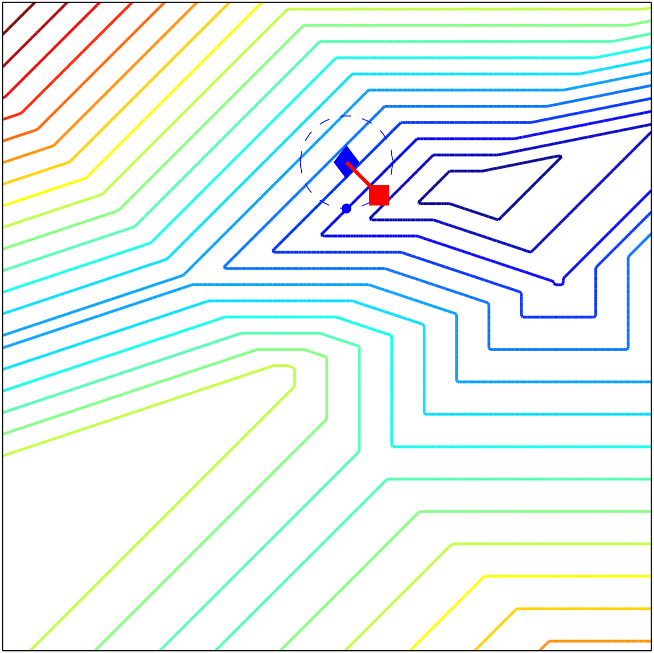


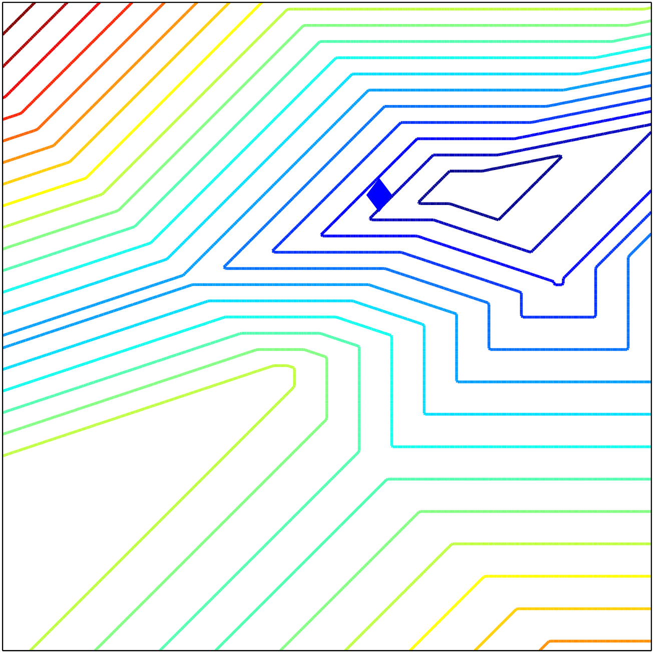


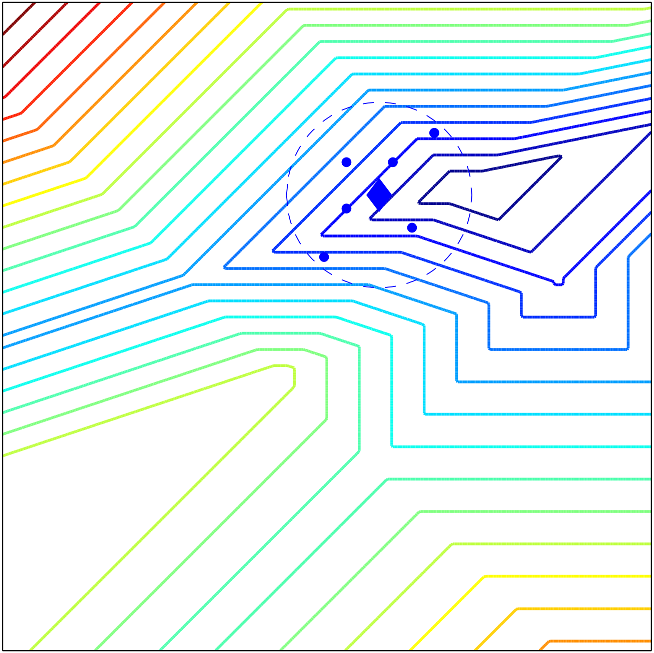


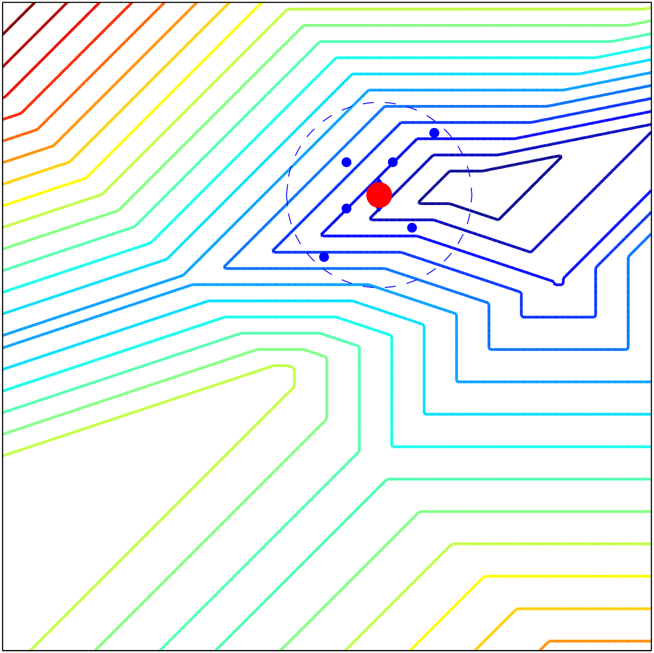


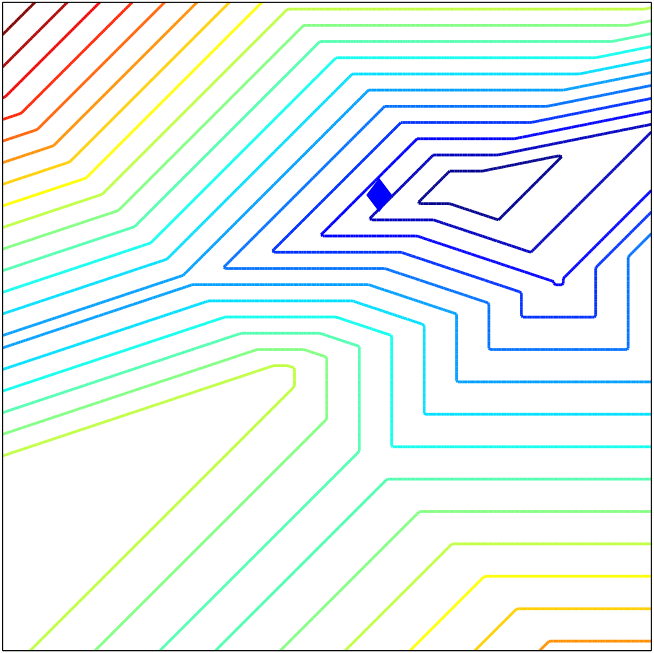


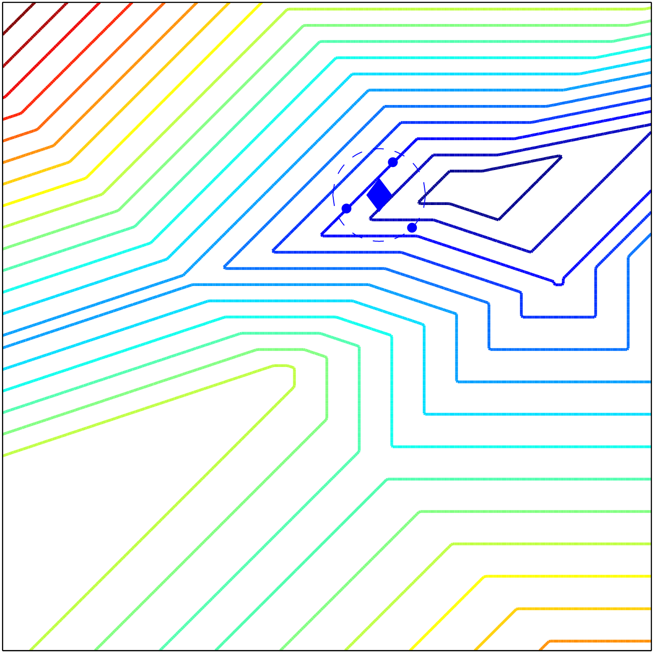


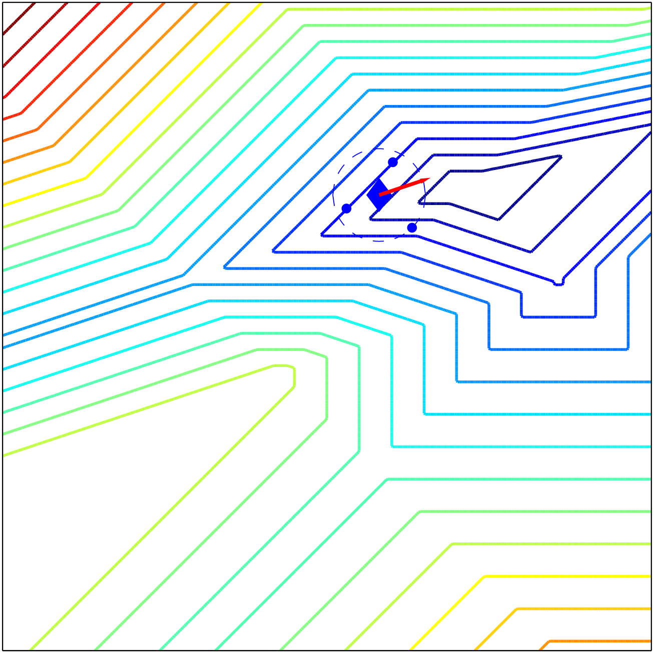


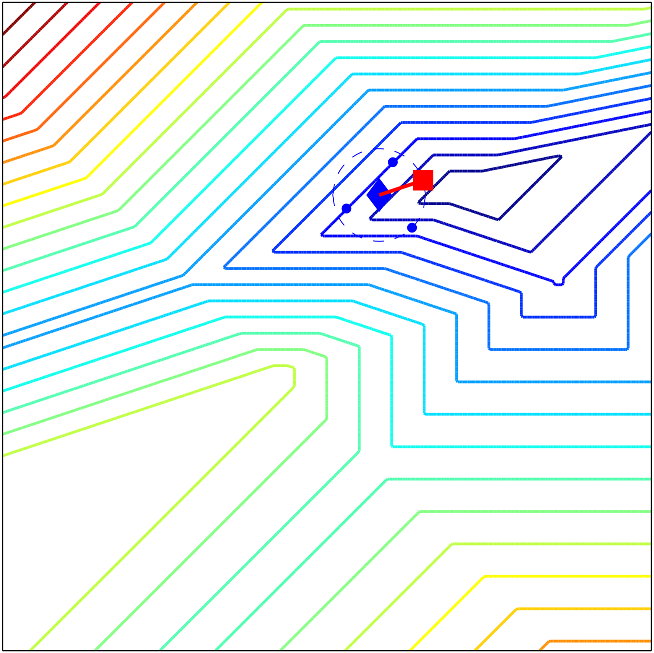


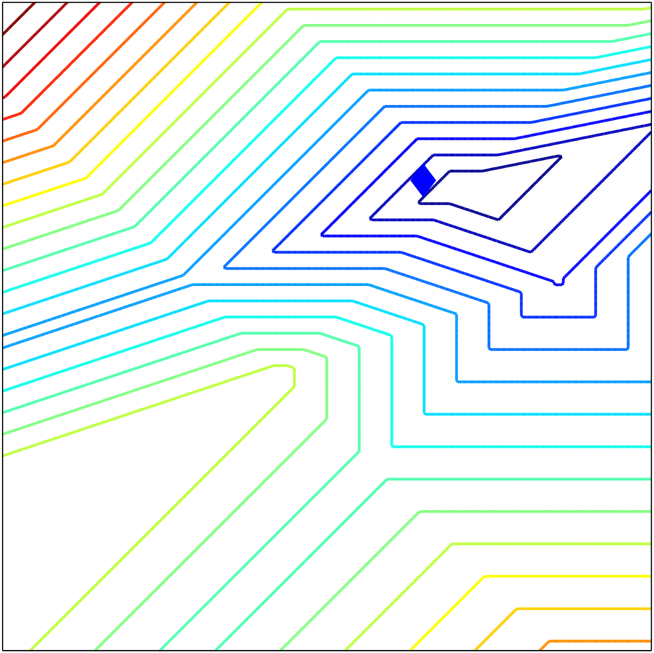


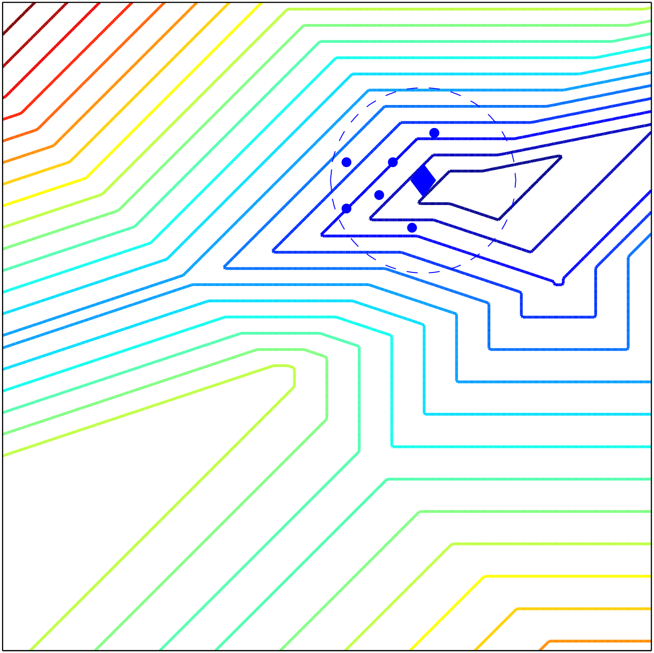


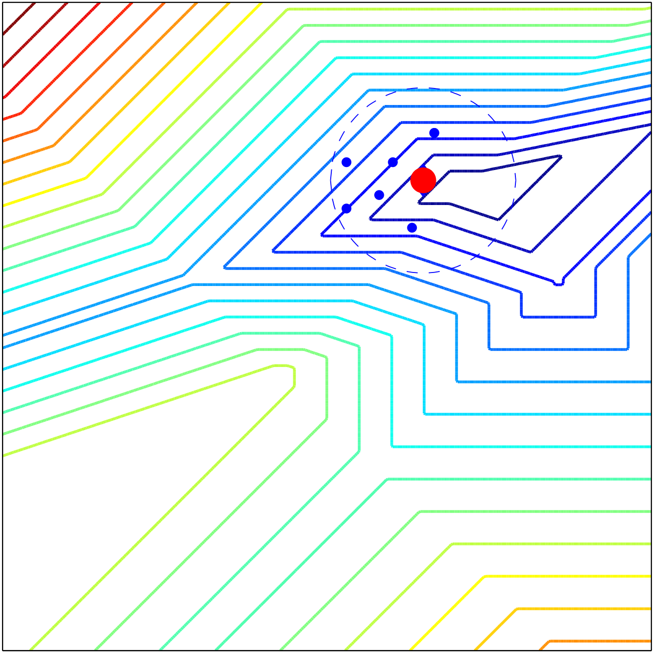


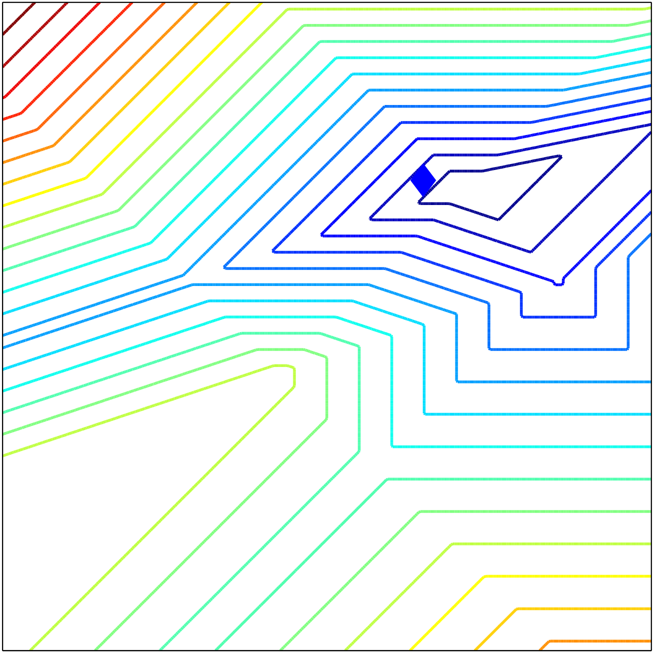


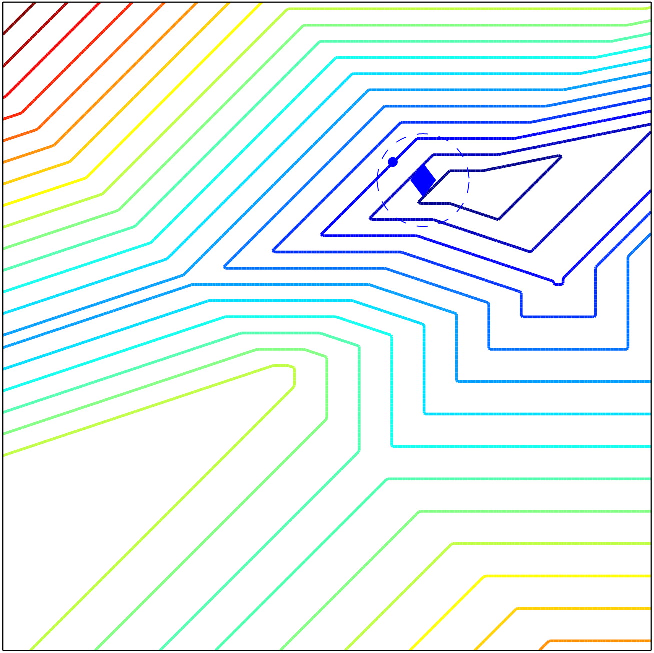


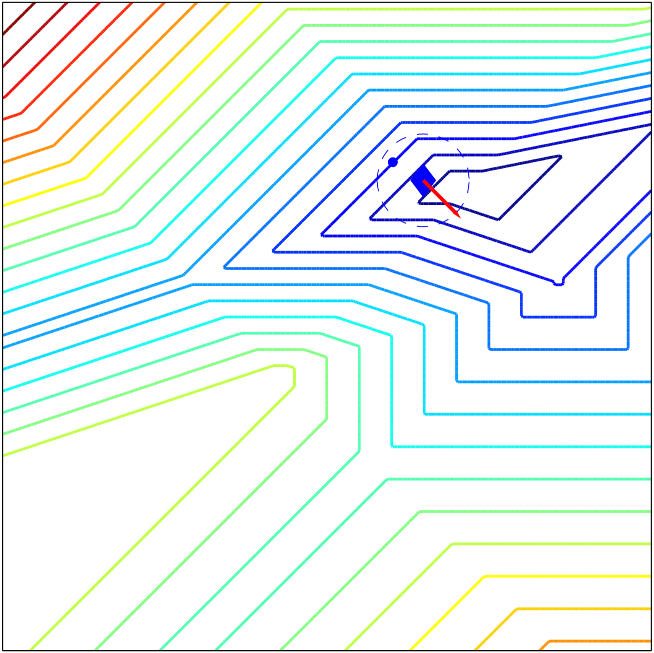


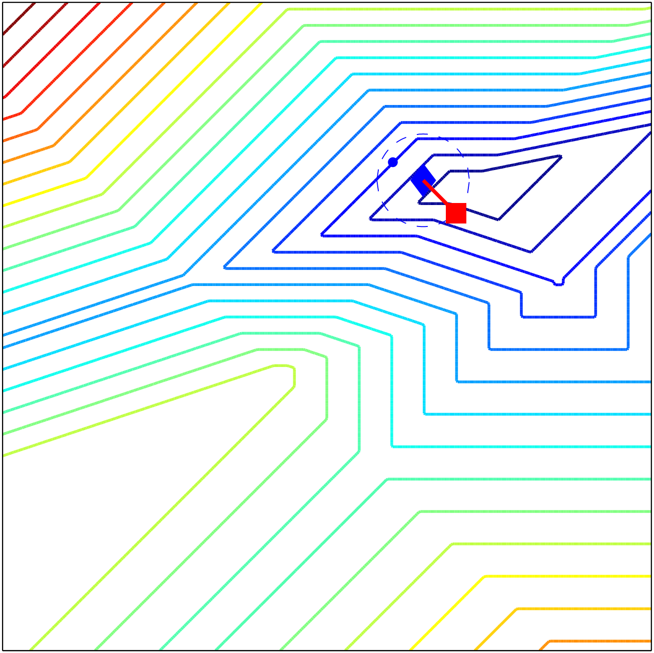


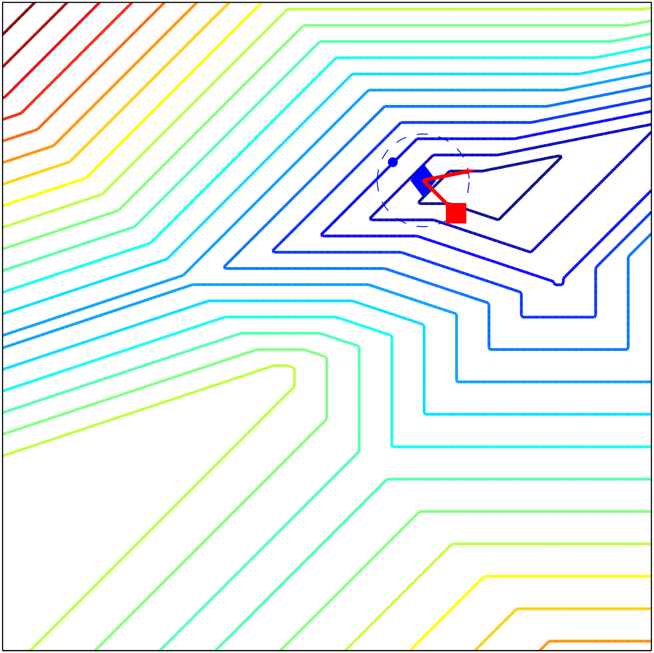


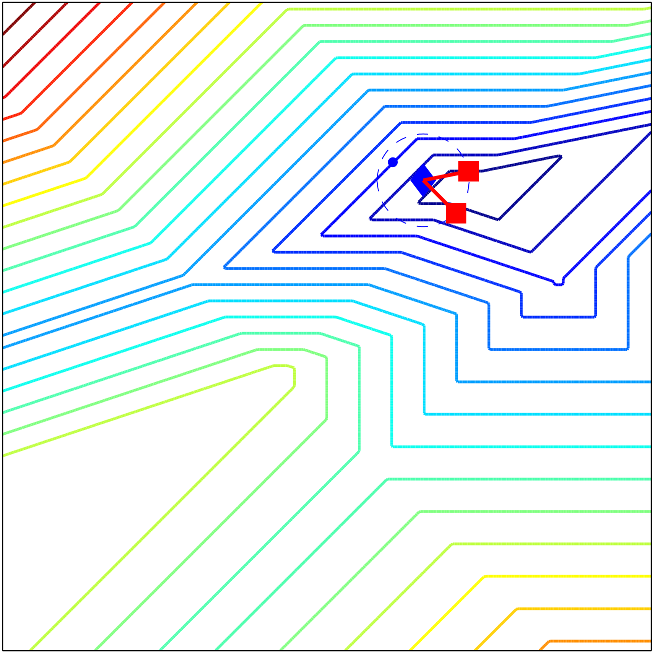


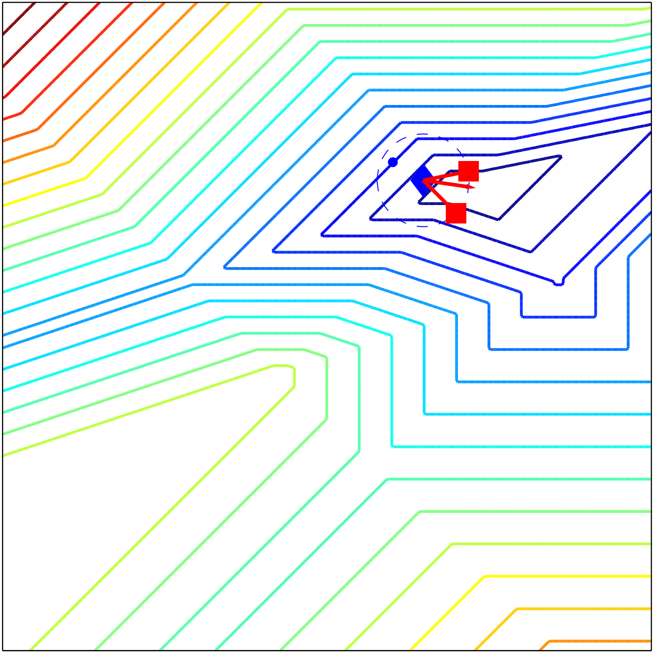


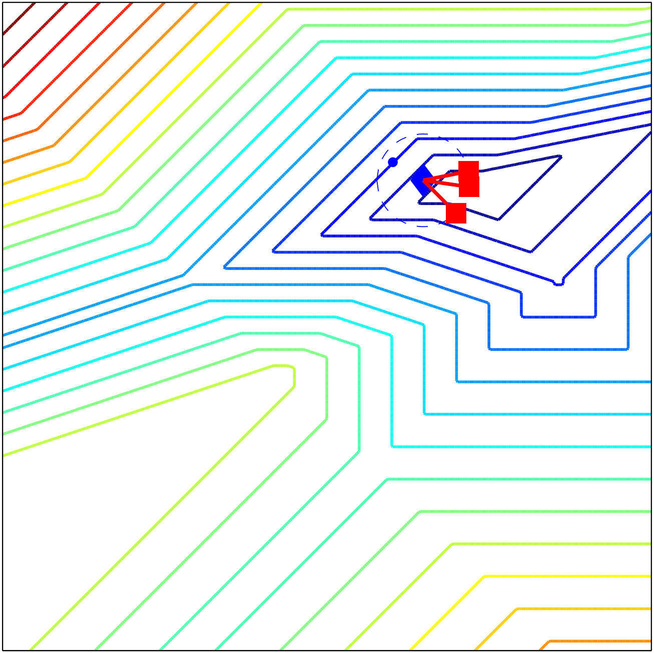


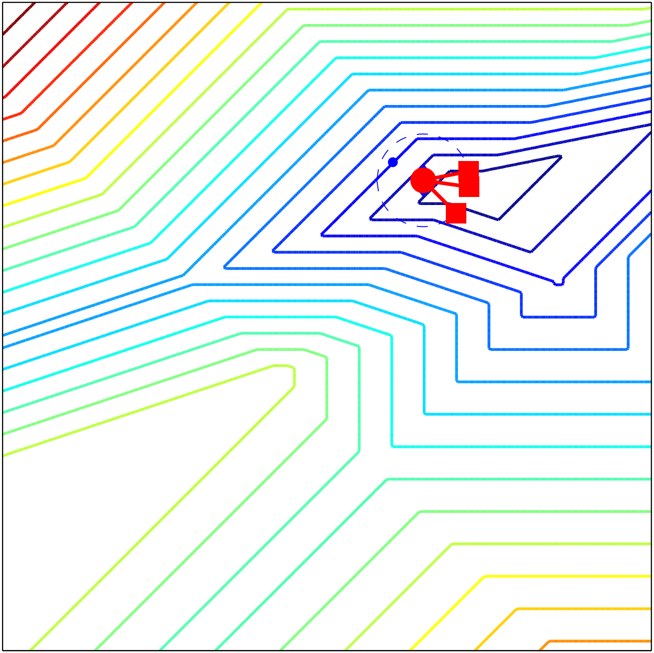


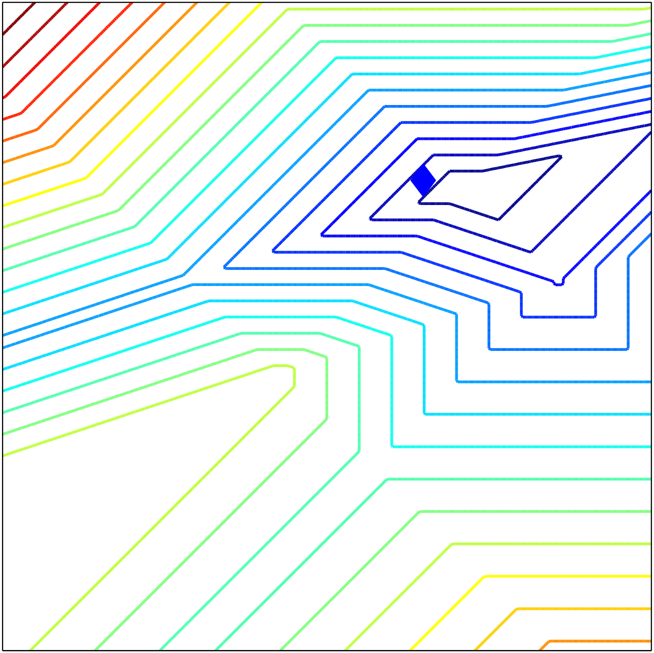


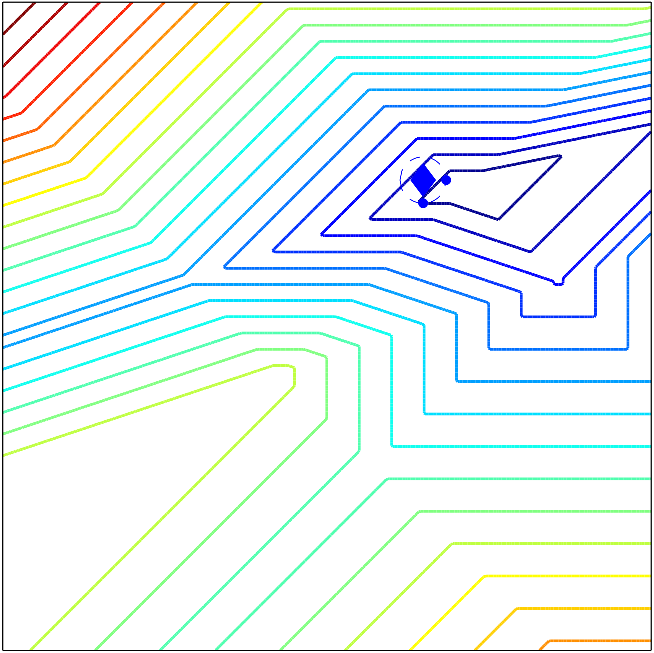


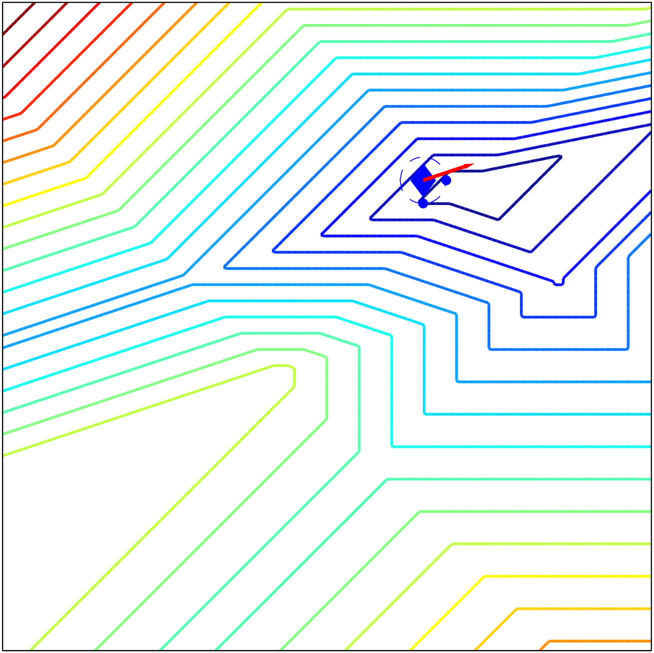


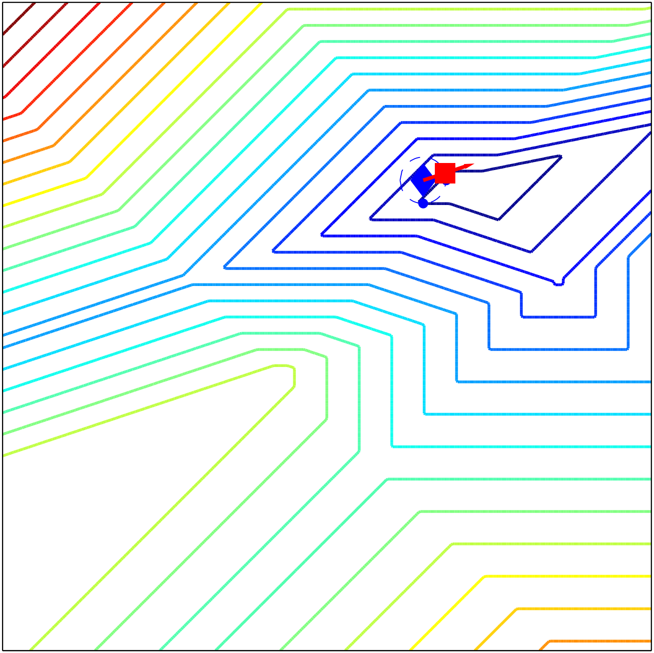


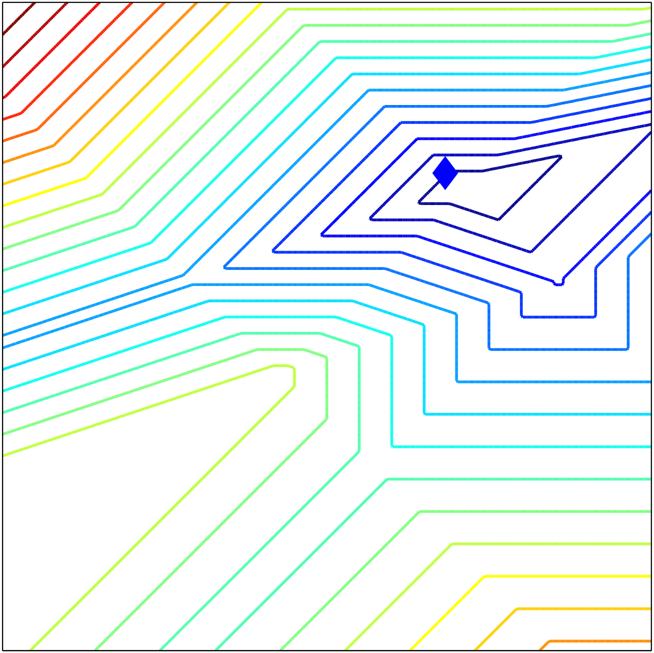


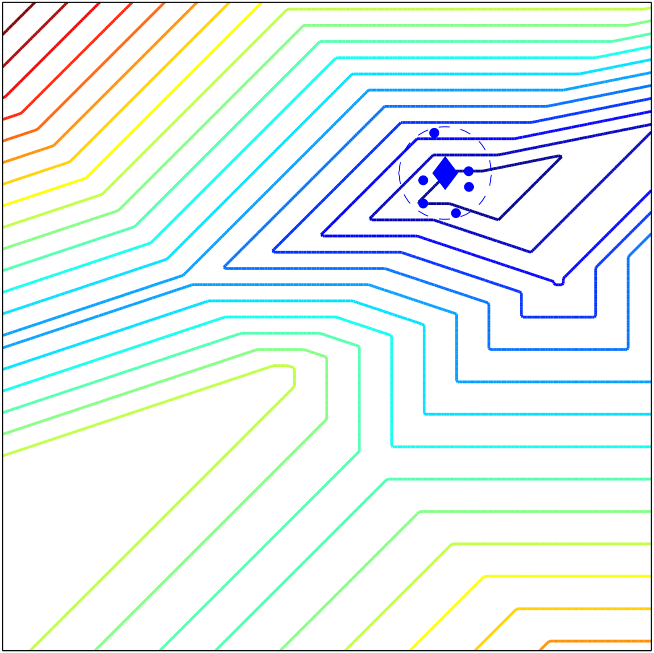


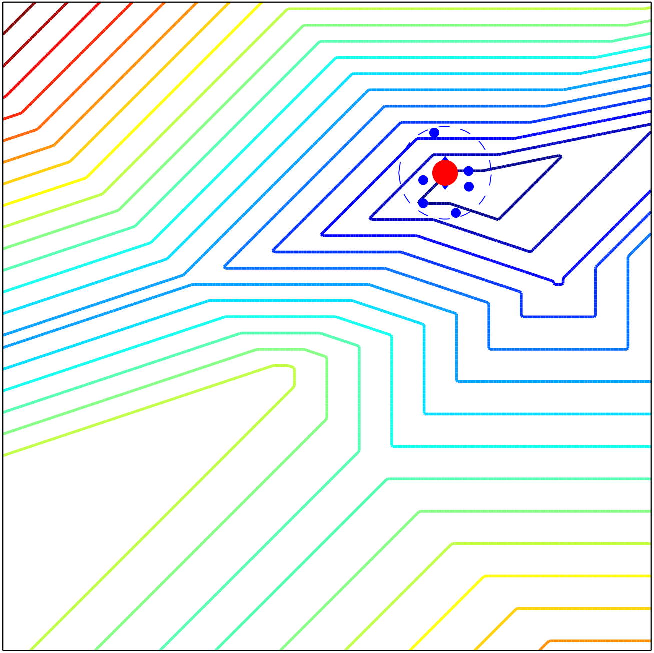












Convergence

- If the trust region radius Δ_k is a sufficiently small multiple of the master model gradient $\left\|g^k\right\|$, the iteration is guaranteed to be successful.

Convergence

- If the trust region radius Δ_k is a sufficiently small multiple of the master model gradient $\left\|g^k\right\|$, the iteration is guaranteed to be successful.

- $\lim_{k\to\infty} \Delta_k = 0$.

Convergence

- If the trust region radius Δ_k is a sufficiently small multiple of the master model gradient $\left\| g^k \right\|$, the iteration is guaranteed to be successful.
- $\lim_{k \to \infty} \Delta_k = 0$.
- Some subsequence of master model gradients g^k goes zero.

Convergence

- If the trust region radius Δ_k is a sufficiently small multiple of the master model gradient $\left\|g^k\right\|$, the iteration is guaranteed to be successful.
- $\lim_{k\to\infty} \Delta_k = 0$.
- Some subsequence of master model gradients g^k goes zero.
- Zero is in the generalized Clarke subdifferential of cluster points of any subsequence of iterates with master model gradients converging to zero.

Convergence

- If the trust region radius Δ_k is a sufficiently small multiple of the master model gradient $\left\|g^k\right\|$, the iteration is guaranteed to be successful.
- $\lim_{k\to\infty} \Delta_k = 0$.
- Some subsequence of master model gradients g^k goes zero.
- Zero is in the generalized Clarke subdifferential of cluster points of any subsequence of iterates with master model gradients converging to zero.
- The same holds for cluster points of the sequence of MS4PL iterates.

Test problems

Let h be a censored ℓ_1-loss function. Given data $d \in \mathbb{R}^p$, censors $c \in \mathbb{R}^p$, and the mapping $F : \mathbb{R}^n \to \mathbb{R}^p$, we define

$$f(x) = \sum_{i=1}^{p} |d_i - \max\{F_i(x), c_i\}| .$$

That is, $\psi = 0$, and

$$h(y) = \sum_{i=1}^{p} |d_i - \max\{y_i, c_i\}| .$$

Test problems

Let h be a censored ℓ_1-loss function. Given data $d \in \mathbb{R}^p$, censors $c \in \mathbb{R}^p$, and the mapping $F : \mathbb{R}^n \to \mathbb{R}^p$, we define

$$f(x) = \sum_{i=1}^{p} |d_i - \max\{F_i(x), c_i\}| .$$

That is, $\psi = 0$, and

$$h(y) = \sum_{i=1}^{p} |d_i - \max\{y_i, c_i\}| .$$

Define F to be the 53 vector mapping in the Móre and Wild benchmarking set. $2 \leq n \leq 12$, $2 \leq p \leq 45$.

Test problems

$$f(x) = \sum_{i=1}^{p} |d_i - \max\{F_i(x), c_i\}|$$

Try to define d and c to introduce many points of nondifferentiability.

Test problems

$$f(x) = \sum_{i=1}^{p} |d_i - \max\{F_i(x), c_i\}|$$

Try to define d and c to introduce many points of nondifferentiability.

Draw c_i from $U(\ell_i, u_i)$

$$\ell_i = \min\left\{F_i(x^0), F_i(x^*)\right\} \qquad \text{and} \qquad u_i = \max\left\{F_i(x^0), F_i(x^*)\right\}.$$

Test problems

$$f(x) = \sum_{i=1}^{p} |d_i - \max\{F_i(x), c_i\}|$$

Try to define d and c to introduce many points of nondifferentiability.

Draw c_i from $U(\ell_i, u_i)$

$$\ell_i = \min\left\{F_i(x^0), F_i(x^*)\right\} \qquad \text{and} \qquad u_i = \max\left\{F_i(x^0), F_i(x^*)\right\}.$$

Make the (crude) assumption that $F_i(x) \sim U(\ell_i, u_i)$, then

$$\max\{c_i, F_i(x)\} \sim (u_i - \ell_i) * \beta(2,1) + \ell_i.$$

Draw d_i from this distribution for $2 \leq i \leq p$.

Test problems

$$f(x) = \sum_{i=1}^{p} |d_i - \max\{F_i(x), c_i\}|$$

Try to define d and c to introduce many points of nondifferentiability.

Draw c_i from $U(\ell_i, u_i)$

$$\ell_i = \min\left\{F_i(x^0), F_i(x^*)\right\} \qquad \text{and} \qquad u_i = \max\left\{F_i(x^0), F_i(x^*)\right\}.$$

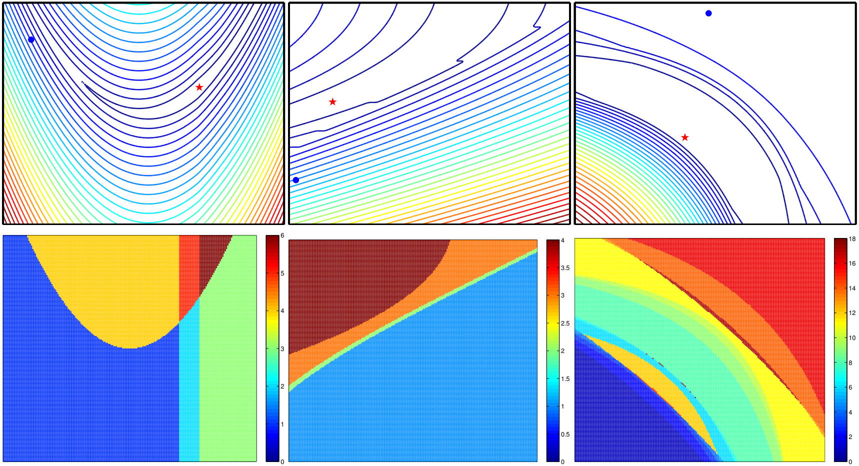
Make the (crude) assumption that $F_i(x) \sim U(\ell_i, u_i)$, then

$$\max\{c_i, F_i(x)\} \sim (u_i - \ell_i) * \beta(2,1) + \ell_i.$$

Draw d_i from this distribution for $2 \leq i \leq p$.

Set $c_1 = -\infty$ and $d_1 = 0$.

Examples



Algorithms to compare

MS4PL-1 Using manifolds at x^k

MS4PL-2 Using manifolds in $\mathcal{B}(x^k, \Delta_k)$

Algorithms to compare

- **MS4PL-1** Using manifolds at x^k
- **MS4PL-2** Using manifolds in $\mathcal{B}(x^k, \Delta_k)$
- **PLC** POUNDERs using a single manifold active at x^k to form a master model

Algorithms to compare

MS4PL-1 Using manifolds at x^k

MS4PL-2 Using manifolds in $\mathcal{B}(x^k, \Delta_k)$

PLC POUNDERs using a single manifold active at x^k to form a master model

SLQP-GS Gradient sampling algorithm from Curtis

GRANSO BFGS-SQP algorithm Mitchell, Curtis, and Overton. (Can handle constraints too.)

Gradient sampling

Theorem (Rademacher)

If $S \subset \mathbb{R}^n$ is open and $f : S \to \mathbb{R}$ is locally Lipschitz on S, then f is differentiable almost everywhere on S.

Gradient sampling

1. Approximate $\partial f(x^k)$ by sampling $m \geq n+1$ points $x^{k,j}$ in $\mathcal{B}(x^k, \epsilon_k)$. Set

$$\mathfrak{G}^k = \operatorname{conv}\left\{\nabla f(x^{k,1}), \ldots, \nabla f(x^{k,m})\right\}$$

Gradient sampling

1. Approximate $\partial f(x^k)$ by sampling $m \geq n+1$ points $x^{k,j}$ in $\mathcal{B}(x^k, \epsilon_k)$. Set

$$\mathfrak{G}^k = \operatorname{conv}\left\{\nabla f(x^{k,1}), \ldots, \nabla f(x^{k,m})\right\}$$

2. Set ξ^k to be the minimum norm element in $\mathfrak{G}^k$.

Gradient sampling

1. Approximate $\partial f(x^k)$ by sampling $m \geq n+1$ points $x^{k,j}$ in $\mathcal{B}(x^k, \epsilon_k)$. Set

$$\mathfrak{G}^k = \operatorname{conv}\left\{\nabla f(x^{k,1}), \ldots, \nabla f(x^{k,m})\right\}$$

2. Set ξ^k to be the minimum norm element in $\mathfrak{G}^k$.
3. Set α_k to be the smallest power s of $\gamma \in (0,1)$ satisfying

$$f(x^k + \gamma^s \xi^k) < f(x^k) - \beta\gamma^s \left\|\xi^k\right\|$$

Gradient sampling

1. Approximate $\partial f(x^k)$ by sampling $m \geq n+1$ points $x^{k,j}$ in $\mathcal{B}(x^k, \epsilon_k)$. Set

$$\mathfrak{G}^k = \operatorname{conv}\left\{\nabla f(x^{k,1}), \ldots, \nabla f(x^{k,m})\right\}$$

2. Set ξ^k to be the minimum norm element in $\mathfrak{G}^k$.
3. Set α_k to be the smallest power s of $\gamma \in (0,1)$ satisfying

$$f(x^k + \gamma^s \xi^k) < f(x^k) - \beta\gamma^s \left\|\xi^k\right\|$$

4. If $\nabla f(x^k + \alpha_k \xi^k)$ exists, $x^{k+1} = x^k + \alpha_k \xi^k$.
   Else, find a point in $\hat{x} \in \mathcal{B}(x^k, \epsilon_k)$ satisfying

$$f(\hat{x}^k + \gamma^s \xi^k) < f(x^k) - \beta\alpha_k \left\|\xi^k\right\|$$

   and set $x^{k+1} = \hat{x}^k + \alpha_k \xi^k$.

Gradient sampling

1. Approximate $\partial f(x^k)$ by sampling $m \geq n+1$ points $x^{k,j}$ in $\mathcal{B}(x^k, \epsilon_k)$. Set
$$\mathfrak{G}^k = \operatorname{conv}\left\{\nabla f(x^{k,1}), \dots, \nabla f(x^{k,m})\right\}$$
2. Set ξ^k to be the minimum norm element in $\mathfrak{G}^k$.
3. Set α_k to be the smallest power s of $\gamma \in (0,1)$ satisfying
$$f(x^k + \gamma^s \xi^k) < f(x^k) - \beta\gamma^s \left\|\xi^k\right\|$$
4. If $\nabla f(x^k + \alpha_k \xi^k)$ exists, $x^{k+1} = x^k + \alpha_k \xi^k$.
   Else, find a point in $\hat{x} \in \mathcal{B}(x^k, \epsilon_k)$ satisfying
$$f(\hat{x}^k + \gamma^s \xi^k) < f(x^k) - \beta\alpha_k \left\|\xi^k\right\|$$
   and set $x^{k+1} = \hat{x}^k + \alpha_k \xi^k$.

- Iterates must not be points of nondifferentiability

Gradient sampling

1. Approximate $\partial f(x^k)$ by sampling $m \geq n+1$ points $x^{k,j}$ in $\mathcal{B}(x^k, \epsilon_k)$. Set

$$\mathfrak{G}^k = \operatorname{conv}\left\{\nabla f(x^{k,1}), \ldots, \nabla f(x^{k,m})\right\}$$

2. Set ξ^k to be the minimum norm element in $\mathfrak{G}^k$.
3. Set α_k to be the smallest power s of $\gamma \in (0,1)$ satisfying

$$f(x^k + \gamma^s \xi^k) < f(x^k) - \beta\gamma^s \left\|\xi^k\right\|$$

4. If $\nabla f(x^k + \alpha_k \xi^k)$ exists, $x^{k+1} = x^k + \alpha_k \xi^k$.
   Else, find a point in $\hat{x} \in \mathcal{B}(x^k, \epsilon_k)$ satisfying

$$f(\hat{x}^k + \gamma^s \xi^k) < f(x^k) - \beta\alpha_k \left\|\xi^k\right\|$$

   and set $x^{k+1} = \hat{x}^k + \alpha_k \xi^k$.

- Iterates must not be points of nondifferentiability
- Significant sampling may be required

Tests

f test A method s solves a problem p to a level τ after j function evaluations if

$$f(x^0) - f(x^j) \geq (1-\tau)(f(x^0) - \tilde{f}_p)$$

x^0 is the problem's starting point, and $\tilde{f}_p$ is the best-found function value.

Tests

f test A method s solves a problem p to a level τ after j function evaluations if

$$f(x^0) - f(x^j) \geq (1-\tau)(f(x^0) - \tilde{f}_p)$$

x^0 is the problem's starting point, and $\tilde{f}_p$ is the best-found function value.

$\partial_{\mathrm{C}} f$ test Sample gradients.

Tests

f test A method s solves a problem p to a level τ after j function evaluations if

$$f(x^0) - f(x^j) \geq (1-\tau)(f(x^0) - \tilde{f}_p)$$

x^0 is the problem's starting point, and $\tilde{f}_p$ is the best-found function value.

$\partial_{\mathrm{C}} f$ test Sample gradients.

Draw 30 points uniformly from $B(x^j, 10^{-8})$ for each point x^j evaluated by each method.

Tests

f test A method s solves a problem p to a level τ after j function evaluations if

$$f(x^0) - f(x^j) \geq (1-\tau)(f(x^0) - \tilde{f}_p)$$

x^0 is the problem's starting point, and $\tilde{f}_p$ is the best-found function value.

$\partial_{\mathrm{C}} f$ test Sample gradients.

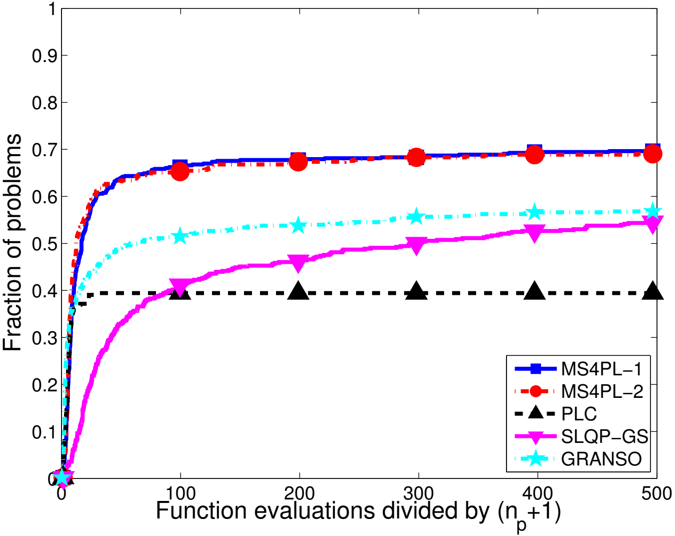
Draw 30 points uniformly from $B(x^j, 10^{-8})$ for each point x^j evaluated by each method.

s solves p to a level τ after j function evaluations if

$$\left\|\tilde{g}^j\right\| \leq \tau \left\|\tilde{g}^0\right\|$$

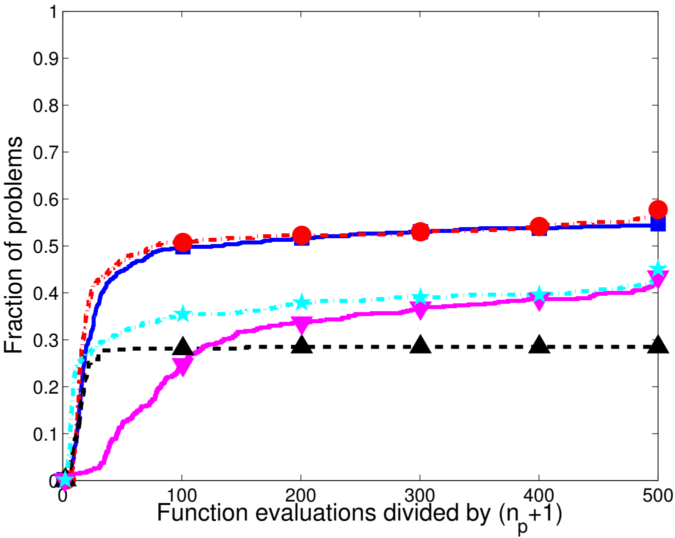
Data profiles

f-test, $\tau = 10^{-2}$

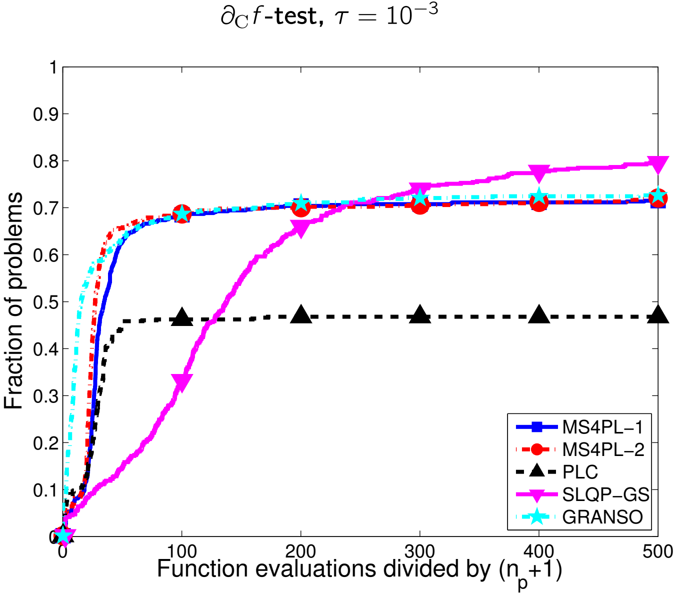


Data profiles

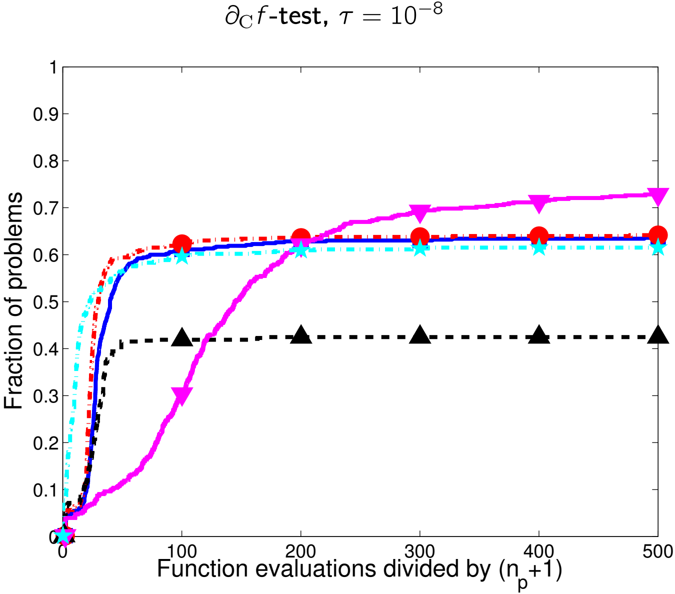
f-**test**, $\tau = 10^{-5}$



Data profiles



Data profiles



Conclusions

When optimizing functions of the form $h(F(x))$ when

- h is "easy"
- F is "hard"

it can be advantageous to model F_i and then combine those models via known information about h.

Conclusions

When optimizing functions of the form $h(F(x))$ when

- h is "easy"
- F is "hard"

it can be advantageous to model F_i and then combine those models via known information about h.

Email `jmlarson@anl.gov` for a preprint.

Thank you!